



Universidad Carlos III de Madrid

Escuela Politécnica Superior

TRABAJO FIN DE GRADO

**Aplicación Web para el seguimiento de Candidatos y Control de
Indicadores Claves de Rendimientos en Procesos de Selección
de Recursos Humanos**

Raúl de la Torre Navarro

Grado en Ingeniería de Sistemas Audiovisuales

Tutor: José María Rubio Manso

Septiembre 2017

AGRADECIMIENTOS

Empecemos por lo personal, gracias a la familia, padre, madre y hermano. Gracias por aguantar mis cambios de humor, gracias por pagar la “fiesta” de la universidad, no precisamente barata. Matrículas astronómicas han tenido que pagar.

En lo docente, pocos profesores a destacar en esta travesía que supone la universidad. Sobre todo, nombrar a mi tutor de este proyecto, por su atención, comprensión y sobre todo y lo más importante por utilizar su tiempo en ayudarme a mejorar, sobre todo en la realización de dicho trabajo.

En lo profesional, nombrar a la última empresa donde trabaje, The Key Talent. Básicamente, me enseñaron que existe mundo fuera de la ingeniería y que no somos el ombligo de nada, me dieron la visión multidisciplinar y polivalente que tengo.

También agradecer a mi yo pasado el no haber abandonado hasta llegar al final de esta etapa. Gracias por aguantar tantas zancadillas y tantos inconvenientes, mejorar mi capacidad de reponerme a los problemas, tanto en lo académico, como lo profesional, así como lo personal.

RESUMEN

Vivimos en un mundo de cambios constantes y cada vez más rápidos, sobre todo el mundo de los recursos digitales, con avances cada vez más sustanciales. Sí bien la tercera revolución industrial o “la revolución digital” está catalogada en la época de 1950-1970 por la introducción del uso de computadoras, registros digitales y los cambios radicales que esto supuso. Hoy en día ese cambio se está viendo exponencialmente debido a que la mayor parte de la población puede acceder a estas tecnologías con relativa facilidad.

Por tanto, nos encontramos en una época en la que se potencia el *software* como un servicio para el usuario, facilitando tareas cotidianas ya sea de su vida personal o laboral. Realizando lo que antes costaba un aprendizaje mucho más a fondo sobre el tema y donde una tarea te podía tomar horas para pasar a realizarla en minutos.

Es aquí donde cobrará vital importancia el *software* aplicado al mundo de los RRHH, un sector que actualmente se está digitalizando y creciendo, haciendo que las compras de *software* de RRHH se dispare sustancialmente como se podrá comprobar en el siguiente enlace ("Las ventas del *software* RRHH crecerán entre un 10 y 20%» MuyCanal", 2017).

Actualmente a pesar de ser un sector en continuo cambio nos encontramos con la necesidad de las empresas de tener a su disposición un *software* eficiente que les permita realizar las tareas que llevan realizando con Excel desde hace años pasando a poder realizar sus metodologías de trabajo de forma online, pudiendo trabajar un equipo en conjunto sin miedo a pisar el trabajo de los compañeros. Los consultores de RRHH necesitan una herramienta versátil que les permita un rápido *ranking* y filtrado de candidatos, pudiendo realizar las tareas que antes suponían horas en minutos.

Estamos en la época de potenciar el *software* para el usuario/consultor de RRHH, es más, irá avanzando aún más con el paso de los años, introduciendo nuevas tecnologías aplicadas a las metodologías seguida por los consultores, como por ejemplo, el uso de realidad virtual en un futuro para las pruebas que realizan a los candidatos de un proceso de selección o para los participantes de un proceso de evaluación. Hablaremos de ello en detalle en las posibles implementaciones de trabajos futuros. Todo llega.

ÍNDICE GENERAL

AGRADECIMIENTOS	3
RESUMEN	4
ÍNDICE GENERAL	5
ÍNDICE DE FIGURAS	8
ÍNDICE DE TABLAS	10
Capítulo 1. Introducción	11
1.1. Motivación	11
1.2. Objetivos	12
1.3. Estructura de la memoria	12
Capítulo 2. Planteamiento del problema	14
2.1. ¿Qué es una solución ATS?	14
2.2. Estado del arte	15
2.2.1. FileFinder	15
2.2.2. Zoho Recruit	17
2.2.3. Bullhorn	19
2.2.4. Oracle Taleo Cloud Service	20
2.2.5. Greenhouse	21
2.2.6. NewtonSoftware	22
2.2.7. Talent Clue	22
2.2.8. Comparativa genérica de soluciones	25
2.3. Tecnologías utilizadas	26
2.3.1. HTML5 y CSS3	26
2.3.2. JavaScript	27
2.3.3. jQuery	28
2.3.4. Python	28
2.3.5. SQL	29
2.3.6. JSON	30
2.3.7. SublimeText 3	31
2.3.8. Trello	32
Capítulo 3. Metodología de Trabajo (Scrum)	35
3.1. Historia	35
3.2. Introducción	35
3.3. Descripción General de la Metodología	36
3.3.1. Fundamentación	36
3.3.2. Valores de trabajo	36

3.3.3. Roles del proyecto	37
3.4. Requisitos de nuestro sistema	37
Capítulo 4. Análisis del Sistema	41
4.1. Product Backlog	42
4.1.1. Plan de producto	42
4.1.1.1. Responsabilidades de los implicados	42
4.2.1. Responsabilidades de los implicados en el Product Backlog	45
4.2.2. Planificación Sprint	46
4.3. Sprint Backlog	47
4.3.1. Responsabilidades de los implicados	48
4.3.2. Primer Release/Sprint	48
4.3.2.1. Incremento	51
4.3.3. Segundo Release/Sprint	52
4.3.3.1. Incremento	56
4.3.4. Tercer Release/Sprint	57
4.3.4.1. Incremento	60
4.3.4.2. Gráfica de Avance (Burn Down)	61
4.3.4.2.1 Responsabilidades de los implicados	61
4.4. Otros puntos de interés sobre la metodología utilizada	62
4.4.1. Reunión de inicio de sprint	62
4.4.2. Reunión técnica diaria	63
4.4.3. Reunión de cierre de sprint y entrega de incremento	64
Capítulo 5. Diseño de la solución técnica	65
5.1. Bases de datos	67
5.2. Google Analytics	69
5.3. Formulario de inscripción	71
5.4. Dashboard	72
5.4.1. Widget – Fases del proyecto	72
5.4.2. Widgets – KPI's	74
5.4.3. Tabla con información	74
5.5. Interfaz de usuario	77
5.5.1. Estudio de diseño	77
5.5.2. Landing Page y formulario	77
5.5.3. Dashboard	79
Capítulo 6. Pruebas de los sistemas y manual de usuario	80
6.1. Pruebas de la aplicación	80

6.2. Manual de usuario	83
6.2.1. Landing Page	84
6.2.2. Formulario	85
6.2.3. Página de dashboard	86
Capítulo 7. Presupuesto	89
Capítulo 8. Conclusiones y futuras entregas	93
8.1. Conclusiones	93
8.2. Trabajos Futuros	94
8.2.1. Primer trabajo futuro	94
8.2.2. Segundo trabajo futuro	95
8.2.3. Tercer trabajo futuro	95
Referencias	97
SUMMARY	100

ÍNDICE DE FIGURAS

Figura 1. Look & Feel aplicación	16
Figura 2. Look & Feel aplicación	18
Figura 3. Look & Feel de candidatos	18
Figura 4. Look & Feel de Bullhorn	19
Figura 5. Look & Feel de Candidaturas	20
Figura 6. Look & Feel de información del candidato	21
Figura 7. Look & Feel de Landing Page	23
Figura 8. Look & Feel de Oferta	24
Figura 9. Logo HTML5 y CSS3	27
Figura 10. Logo de JavaScript	28
Figura 11. Logo de jQuery	28
Figura 12. Logo de Python	29
Figura 13. Logo de Oracle SQL Developer	30
Figura 14. Logo JSON	31
Figura 15. Ejemplo de paquetes instalados	32
Figura 16. Logo de Sublime Text 3	32
Figura 17. Uso de Trello	34
Figura 18. Logo de Trello	34
Figura 19. Épica (I)	38
Figura 20. Épica (II)	38
Figura 21. Épica (III)	39
Figura 22. Épica (IV)	39
Figura 23. Épica (V)	40
Figura 24. Épica (VI)	40
Figura 25. Tareas primer release	49
Figura 26. Figura de aviso de cookies	52
Figura 27. Tareas segundo release	53
Figura 28. Imagen de formulario completo	56
Figura 29. Tareas tercer release	57
Figura 30. Gráfica de Burn Down	62
Figura 31. Arquitectura técnica	65
Figura 32. Posible workflow de trabajo	66
Figura 33. Base de datos utilizada	67
Figura 34. Ejemplo de claves en Base de Datos. Models.py	68
Figura 35. Ejemplo de claves de Base de Datos. Fichero.csv	69
Figura 36. Código GA en las cabeceras de Landing Page y formulario	70
Figura 37. Código Ga en los distintos botones	70
Figura 38. Código widgets del formulario	71
Figura 39. Widget - Fases del proyecto	72
Figura 40. Definición de fases del proyecto	73
Figura 41. Barra de tiempo	74
Figura 42. Ejemplo de tabla	75
Figura 43. PopUp con el resto de información sobre el candidato registrado	76
Figura 44. Vista para la acción eliminar a un candidato	76
Figura 45. Botones de RRSS	78
Figura 46. Ir a formulario de inscripción	78
Figura 47. Botón para Aplicar	78
Figura 48. Información sobre registro realizado con éxito	79
Figura 49. Procesos a desarrollar para la usabilidad	80
Figura 50. Botones de RRSS	84
Figura 51. Aceptación y más información de cookies	84

Figura 52. Información relevante sobre el proceso	85
Figura 53. Ir a formulario de inscripción	85
Figura 54. Mensaje de introducción de email no valido	85
Figura 55. Botón para aplicar al proceso	86
Figura 56. Registro realizado con éxito	86
Figura 57. Timeline de proyecto	87
Figura 58. Campo email para poder enviar correos a los candidatos	87
Figura 59. Acciones a realizar	88
Figura 60. Eliminar datos de candidato	88

ÍNDICE DE TABLAS

Tabla 1. Comparativa del estado del arte	26
Tabla 2. Plan de Producto	45
Tabla 3. Plantilla Historia de Usuario	50
Tabla 4. Historia de Usuario (I)	50
Tabla 5. Historia de Usuario (II)	50
Tabla 6. Historia de Usuario (III)	51
Tabla 7. Historia de Usuario (IV)	54
Tabla 8. Historia de Usuario (V)	54
Tabla 9. Historia de Usuario (VI)	55
Tabla 10. Historia de Usuario (VII)	55
Tabla 11. Historia de Usuario (VIII)	55
Tabla 12. Historia de Usuario (IX)	59
Tabla 13. Historia de Usuario (X)	59
Tabla 14. Historia de usuario (XI)	60
Tabla 15. Historia de Usuario (XII)	60
Tabla 16. Historia de Usuario (XIII)	60
Tabla 17. Plantilla de pruebas	81
Tabla 18. Prueba (I)	81
Tabla 19. Prueba (II)	81
Tabla 20. Prueba (III)	82
Tabla 21. Prueba (IV)	82
Tabla 22. Prueba (V)	82
Tabla 23. Prueba (VI)	82
Tabla 24. Prueba (VII)	82
Tabla 25. Prueba (VIII)	83
Tabla 26. Prueba (IX)	83
Tabla 27. Prueba (X)	83
Tabla 28. Prueba (XI)	83
Tabla 29. Presupuesto primer release	90
Tabla 30. Presupuesto segundo release	90
Tabla 31. Presupuesto tercer release	90
Tabla 32. Margen de beneficio de los tres release	90
Tabla 33. Margen de riesgo de los tres release	91
Tabla 34. Cantidad total primer release	91
Tabla 35. Cantidad total segundo release	91
Tabla 36. Cantidad total tercer release	92

Capítulo 1. Introducción

1.1. Motivación

Todos lo hemos visto, cuando realizamos una entrevista de trabajo, cuando RRHH nos hace una evaluación de seguimiento, cuando participamos en un proceso de formación en la que interviene RRHH para verificar nuestro progreso y como no, comprobar que el dinero que están invirtiendo en nosotros está teniendo un uso práctico, hemos visto como los consultores apuntan en una hoja de Excel los parámetros oportunos a medir, incluso a veces en una hoja de papel. Como futuro ingeniero que vive en el año 2017 le duele ver estos actos, personas que no sacan todo su potencial pues no usan los recursos disponibles al alcance de su mano para automatizar la mayor parte de sus tareas.

Evidentemente estamos ante un inconveniente, todo tiene un precio, para eso se estudia la forma más óptima de conseguir lo que buscamos realizando los estudios pertinentes y maquetando un posible modelo que sirva para la gran mayoría de casos posibles, reduciendo así en gran medida los costes.

Estos hechos que relatamos con anterioridad suponen un empuje para la realización de este proyecto, ayudando a cambiar el mundo, consiguiendo pasar de lo “analógico” a lo digital, entrecomillo analógico porque en realidad no es analógico, siendo serios sería más un mundo sin automatizar los procesos oportunos. Añadir, que tristemente cuando un ingeniero se gradúa el mundo de los RRHH es un gran desconocido para él, suponiendo así un reto este proyecto en la definición de tareas, preguntas a realizar a los candidatos/participantes, así como la arquitectura y diseño de la aplicación para el consultor, haciendo posible un interfaz de usuario amigable para este.

El *software* en el mundo de los RRHH es un medio de trabajo que debería de ser imprescindible, es más, el *software* debe de ser una página web donde los consultores trabajen, no una aplicación de escritorio que dependa también de los requisitos de la potencia de nuestro equipo. Estamos ante la necesidad de trabajar desde cualquier dispositivo sin vernos obligados a tener unos requisitos mínimos para su uso, siendo única y exclusivamente necesario tener conexión a internet y un navegador desde dónde poder trabajar.

1.2. Objetivos

En el presente trabajo fin de grado se deberá lograr los siguientes objetivos generales:

1. Desarrollo de una plataforma que mejore la eficiencia y accesibilidad de los procesos de selección de los candidatos.
2. Integración de información relativa a los indicadores del proceso.

Tendremos así los siguientes objetivos específicos que irán alineados con los objetivos generales:

1. Web accesible, multidispositivo y usable para los usuarios.
2. Integración de *feedback* continuado para los candidatos.
3. Capacidad de mostrar el seguimiento de los candidatos durante el proceso.
4. Integración para poder realizar informes de forma online desde la plataforma. Facilitando el trabajo a los consultores de RRHH.
5. Integración de unas pruebas de evaluación para evaluar el desempeño y las competencias de los participantes en los distintos procesos.

Estos objetivos se irán alcanzando según vayamos implementando nuevas funcionalidades en el tiempo. Así no llegaremos a tener todos los objetivos deseables por una cuestión de tiempo, pero sí en un futuro próximo.

1.3. Estructura de la memoria

La presente memoria del trabajo de fin de grado se desglosa en los siguientes capítulos:

- Capítulo 1: Se recoge la motivación del proyecto y los objetivos que pretendemos alcanzar con este, así como la estructura de la memoria.
- Capítulo 2: Planteamiento del problema, así como el estado actual del mercado de los RRHH.
- Capítulo 3: Planteamiento de la metodología de trabajo, así como un poco sobre su historia.

- Capítulo 4: Análisis del sistema, haciendo hincapié en el *product backlog*, así como el *sprint backlog* y otros puntos/reuniones importantes sobre la metodología para el análisis del sistema.
- Capítulo 5: Diseño de la solución técnica, dónde detallamos lo utilizado para la realización de diferentes *releases*.
- Capítulo 6: Pruebas realizadas sobre los sistemas de nuestra aplicación y un breve y descriptivo manual de uso para los usuarios, manual a ampliar según vaya ampliándose funcionalidades.
- Capítulo 7: Explicamos el presupuesto de nuestro proyecto.
- Capítulo 8: Conclusiones de nuestro proyecto y breve explicación de las épicas para futuros incrementos e implantación de nuevas funcionalidades.

Capítulo 2. Planteamiento del problema

Describiremos con detalle el formato, la temática y el estado del arte de la aplicación desarrollada.

Aquí recogeremos las bases teóricas que engloban el proyecto, resumimos las tecnologías usadas y una breve explicación de estas, así como las oportunas aplicaciones usadas para la gestión de la realización del trabajo fin de grado utilizadas en conjunto con José María.

2.1. ¿Qué es una solución ATS?

A continuación, vamos a explicar de manera introductoria que son y para que sirven las soluciones de *software* ATS (*Applicant Tracking System*).

“Cada vez son más utilizados. Los '*Applicant Tracking Systems*' -o ATS- son programas informáticos capaces de jerarquizar a los candidatos de un proceso de selección de forma automática. Este tipo de '*software*' busca y filtra en la base de datos algunos parámetros y palabras clave para seleccionar los perfiles más adecuados para cada oferta. De esta manera, la empresa puede acceder a los mejores currículums sin tener que leer todos los documentos recibidos.” (“Cómo superar los '*Applicant Tracking Systems*' para encontrar empleo", 2017).

Lo cual permitirá a los consultores de RRHH y todo el personal relacionado con los seguimientos de los procesos un trabajo más eficiente y rápido, sin necesidad de ir visualizando uno a uno todos los currículos vitae de los candidatos. También un buen *software* de ATS tendrá la capacidad de ser utilizado para procesos de evaluación, así como los de formación, habrá que definir entonces una arquitectura lo suficientemente flexible como para no realizar los proyectos AD-HOC y que los desarrollos de cada uno de los distintos proyectos sean similares.

Esto último lo veremos por encima cuando hablemos de la arquitectura de nuestra aplicación, hablaremos de las distintas posibilidades que tiene nuestra potencial aplicación.

2.2. Estado del arte

Información sobre distintas empresas que desarrollan *software* ATS ("Results: Applicant Tracking Software", 2017), vemos que sin realizar ningún filtrado obtenemos una búsqueda de 256 sistemas, pero si seleccionamos aun así empresas que son única y exclusivamente agencias de reclutamiento, seguimos teniendo 162 sistemas encontrados. Vemos que la competencia en el mercado es mucha y muy fuerte. Por estos motivos explicaremos en mayor detalle algunas de las soluciones que ofrecen las distintas empresas para así poder posicionar nuestra aplicación respecto a las demás, analizando los distintos pros y contras.

2.2.1. FileFinder

FileFinder ("FileFinder: Executive Search Software | Cloud Executive Recruiting Software | Headhunter Software", 2017), es una solución desarrollada para la búsqueda de candidatos con un perfil ejecutivo que el día de mañana podrían también a su vez convertirse en potenciales clientes. Dicho esto, observamos que es una aplicación con función de CRM, pero al ofrecer infinidad de funciones podría fácilmente utilizarse como una herramienta de reclutamiento.

Veamos más a fondo que nos ofrece este producto:

- Permite trabajar en un entorno virtual desde cualquier equipo y/o móvil.
- Trabajar sin dejar de lado Outlook, pues va integrado con la solución.
- Posibilidad de traer las distintas informaciones de los potenciales candidatos desde redes profesionales como LinkedIn.
- Manejar tus búsquedas y tu desarrollo de negocio pudiendo monitorear estos a través de los distintos indicadores clave.
- Así como realizar informes para el cliente y exportarlos en Word.

Cuando pedimos más información al respecto la aplicación proporciona infinidad de soluciones, siendo muy óptima para las funciones que nosotros queremos desarrollar. Sin

embargo, observamos los siguientes problemas que a día de hoy pueden generar el rechazo de los consultores. Los problemas encontrados son los siguientes:

- La versión web del producto no está completada todavía y no ofrece todas las soluciones que ofrece la aplicación de escritorio, lo que nos lleva a tener un producto que no está 100% disponible en todos los dispositivos.
- Lentitud a la hora de realizar distintos tipos de filtrados para seleccionar al candidato idóneo.
- Para volcar la información desde redes como LinkedIn, primero debemos descargarnos el CV en formato PDF desde el perfil de dicho candidato y después volcarlo a la plataforma, no siempre vuelca los datos y no los “parsea” de forma adecuada, muchas veces el traspaso de datos debe de realizarse a mano, lo que nos genera pérdida de tiempo.
- La más importante de todas, al tratarse básicamente de una aplicación de escritorio nuestros equipos tienen que tener unos requisitos mínimos. El problema de esto no se encuentra en cuanto memoria RAM necesitamos o la velocidad de nuestro procesador, el problema lo encontramos en que necesitamos una resolución mínima de pantalla un poco más grande lo que suelen ofrecer los ordenadores portátiles. Debemos tener en cuenta que la gran mayoría de empresas al entrar lo que te proporcionan es eso, un ordenador portátil.

A continuación, podemos ver una imagen de la aplicación:

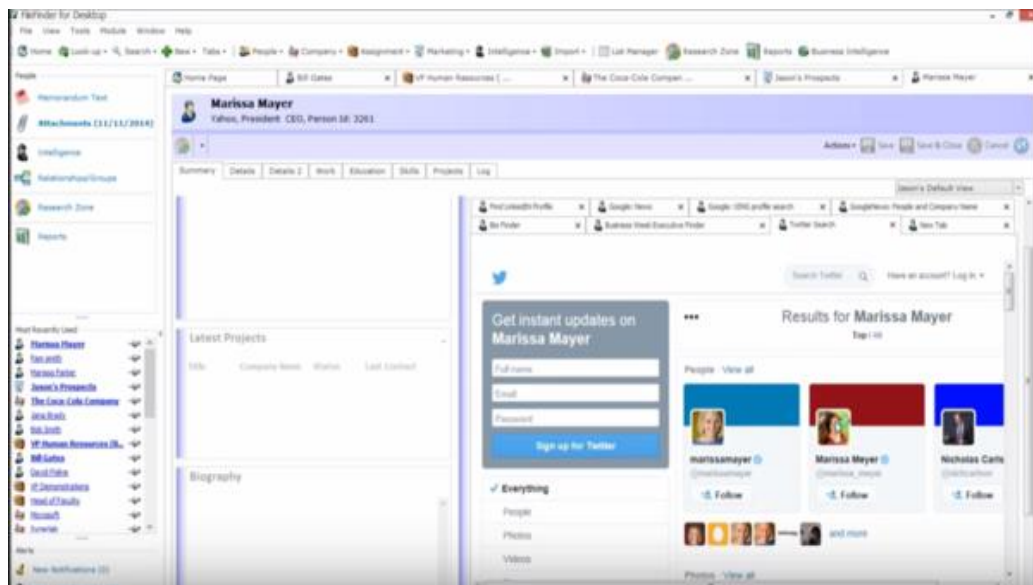


Figura 1. Look & Feel aplicación

Como podemos observar es una aplicación agradable para la vista del usuario y que a pesar de tener muchas funciones podríamos considerar como amigable para el cliente final. Los problemas de esta son los detallados con anterioridad sobre dicha aplicación.

2.2.2. Zoho Recruit

En el siguiente blog donde hablan sobre los productos de Zoho (Pérez, 2017), en este artículo en concreto hablan sobre la última actualización de Zoho Recruit.

Zoho Recruit es una solución que nos permite la automatización, así como integraciones avanzadas con distintas redes sociales, como Facebook y una seguridad óptima para guardar los datos de nuestros candidatos.

Esta plataforma nos permite trabajar desde cualquier dispositivo pues está alojada en la nube y podremos trabajar con ella siempre y cuando tengamos un navegador y conexión a internet.

Cabe destacar, como no, su bajo precio para la contratación y uso de esta plataforma, dependerá del plan que queramos utilizar. Podremos escoger un plan gratuito con unas funciones muy restringidas, pero que si tenemos una empresa muy pequeña tal vez sea más que suficiente y después tendremos distintos tipos de planes a estudiar dependiendo de lo que nosotros queramos tener como funcionalidades para mejorar nuestro flujo de trabajo.

Problemas que nos encontramos con el uso de este *software*.

- Tendremos un interfaz de usuario que deja bastante que desear, aunque los colores podremos personalizarlos, el interfaz parece más de una página que no se ha renovado con el tiempo.
- Las opciones para realizar filtrados de candidatos son escuetas, pues dependerá de cómo “costumicemos” nosotros mismo la plataforma, una vez cambiemos de página en el caso de tener muchos candidatos en nuestra plataforma ese filtro desaparecerá y deberemos de volver a aplicarlo.
- Es un producto que te permite la personalización de este, pero hasta cierto punto, pues para mejorar el flujo de trabajo de los consultores aplicando una metodología de trabajo típica necesitaremos realizar muchos desarrollos.
- Los desarrollos son realizados por los distintos partners de Zoho en los distintos países en los que tiene presencia, no obstante, son caros de realizar y no siempre

proporcionan la funcionalidad esperada.

- Respecto a los informes que genera la plataforma si estás realizando un proceso de selección para un cliente, este no querrá informes del tipo que proporciona Zoho, pues los querrá preparados para él y Zoho no nos permite una personalización demasiado flexible para lo que los clientes suelen pedir como informes de candidatos.

Muy importante, para tener en cuenta es que posee la opción de usarlo como una aplicación móvil. Con estos pros y contras se convierte así en una herramienta muy útil si tienes una empresa pequeña o muy pequeña.

A continuación, podremos ver el aspecto de dicha aplicación:

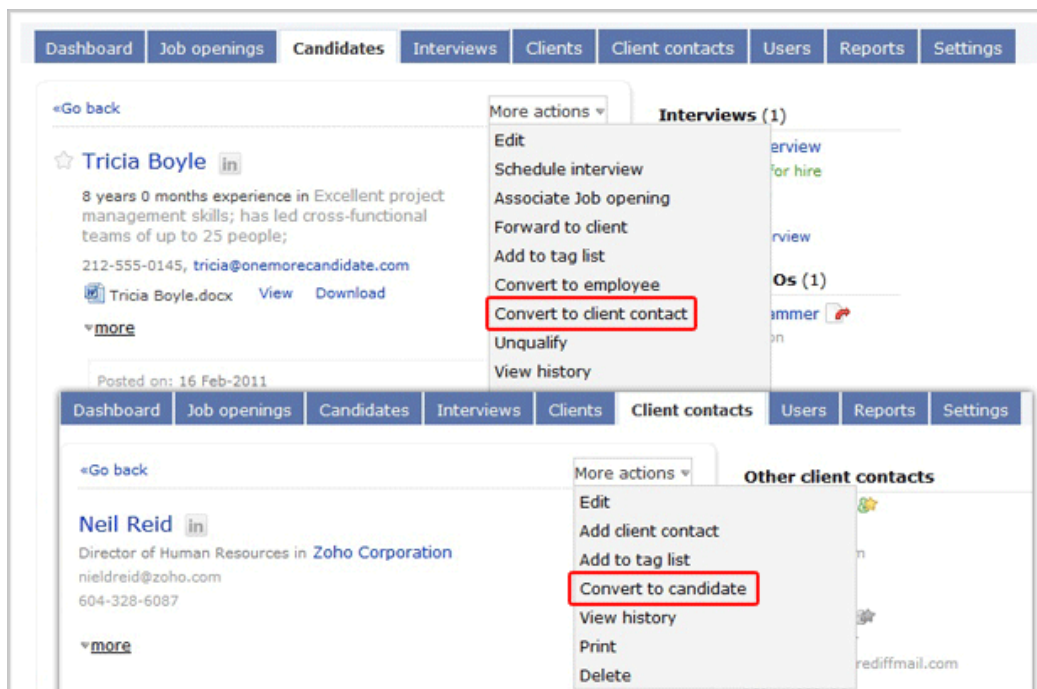


Figura 2. Look & Feel aplicación

Candidate Name	Email	Job	Candidate Status	Updated On	Source	Candidate Owner
John Lee	john.lee@zohorecruit.com	Recruiter	New	21/04/2015 11:35 AM	Internal	Chris Michael
Ganesh Kumar S	ganesh.kumar.s@zohorecruit.com	Recruiter	New	23/02/2015 03:13 PM	Internal	Barish Charles
Tomato CVT	tomato.cvt@zohorecruit.com	Channel	Waiting for Evaluation - Technical Consultant	24/02/2015 04:13 PM	Google	Barish Charles
Sugra International	sugra.international@gmail.com	Acad	Associated -> Java Developer	04/03/2015 17:39 AM	Imported by partner	Barish Charles
James Wilson	james.wilson@zohorecruit.com	Acad	Interview Scheduled -> Business Development Analyst	04/03/2015 05:16 PM	Imported by partner	Barish Charles
John Doe	john.doe@zohorecruit.com	Channel	Interview Scheduled -> Auto Mobile Engg	04/03/2015 02:57 PM	Imported by partner	Barish Charles
Gita Choudhary	gita.choudhary@gmail.com	Channel	Associated -> Associate Manager	24/02/2015 11:42 AM	Imported by partner	Barish Charles
James Hansen	james.hansen@zohorecruit.com	Acad	Associated -> Associate Manager	04/03/2015 04:54 PM	Internal	Barish Charles
Steve Morgan	steve.morgan@zohorecruit.com	Acad	Associated -> Candidate Case Manager	04/03/2015 12:14 PM	Internal	Barish Charles
John Doe	john.doe@zohorecruit.com	Acad	Associated -> Associate	04/03/2015	Internal	Barish Charles

Figura 3. Look & Feel de candidatos

Como observamos, a simple vista se ve que es una herramienta sencilla de utilizar y de gestionar, aunque como mencionábamos su aspecto no parece acorde a lo visto actualmente hoy en día. Además, de los problemas también comentados sobre la falta de flexibilidad a la hora de personalizar dichos puntos críticos para tu empresa.

2.2.3. Bullhorn

Bullhorn ("Staffing Software | Applicant Tracking System | Bullhorn", 2017), donde podremos ver otro ejemplo de una solución que nos proporciona distintos *widgets* según comentan en su página web sobre las analíticas de sus proyectos, así como la mejora en la velocidad de la realización del trabajo de sus clientes. Tratándose de un sistema intuitivo que facilitará la vida a los empleados, podemos observar que posee herramientas de organización, así como un calendario donde se podrán realizar diversas acciones como programar entrevistas, etc.

No obstante, no vemos un sitio donde podamos realizar un filtrado óptimo de nuestros candidatos, así como un seguimiento de las pruebas de selección que puedan realizar y generar un informe de forma automática con todos estos datos.

A continuación, vemos el aspecto de su solución tecnológica

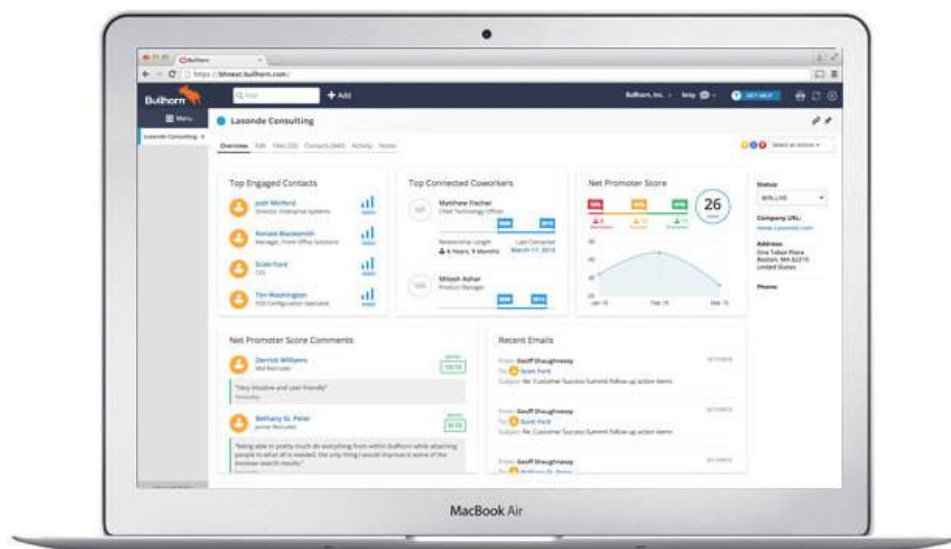


Figura 4. Look & Feel de Bullhorn

No podemos obtener mucha más información sobre el producto sin pedir una Demo de este. No pediremos la Demo por Gestión de tiempos debido a que tenemos muchas posibles soluciones tecnológicas que valorar.

2.2.4. Oracle Taleo Cloud Service

Otra plataforma que hemos encontrado como ATS es la desarrollada por Oracle como es Oracle Taleo Cloud Service ("Oracle Taleo Cloud Service | Cloud Talent | Oracle", 2017).

Por la información obtenida a través de la web y de diferentes vídeos donde podemos ver el funcionamiento de la aplicación, observamos a simple vista que cumple con los requisitos que debe de tener una herramienta de ATS. Pudiendo atraer y retener el talento buscado por las empresas hoy en día, proporcionado analíticas, reclutando desde redes sociales, siendo una aplicación responsiva lo que nos proporcionará la posibilidad de trabajar desde cualquier sitio.

No obstante, no hemos podido testear el *software* haciendo uso de él para tener una apreciación más objetiva sobre este desde el punto de vista del consultor.

Vemos ahora una imagen de cómo es de cara al candidato este *software* en una empresa en la que buscamos empleo. Dicho *software* para los candidatos funciona aplicando en la solución tecnológica de la empresa que ha contratado esta solución y adjuntamos todos los datos relevantes de nuestro perfil en esta. Una vez estamos registrados y con nuestro perfil completo, podremos aplicar a las ofertas de empleo de dicha empresa.



Figura 5. Look & Feel de Candidaturas

En esta imagen vemos el aspecto de cómo ve el candidato las candidaturas a las que ha aplicado para obtener empleo dentro de una empresa.

The screenshot shows a user interface for a job application. At the top, a message states 'Raúl de la Torre Navarro, se ha registrado.' (Raúl de la Torre Navarro, has registered). To the right, there are links for 'Mi cesta de trabajos (0 elementos)' (My job basket (0 items)) and 'Desconectarse' (Log out). Below this, there are two tabs: 'Búsqueda de puestos' (Job search) and 'Mi página de puestos' (My job page), with the latter being active. The main section is titled 'Perfil general' (General profile) and indicates 'Paso 4 de 4' (Step 4 of 4). It features a progress bar with four steps: 'Bienvenid@' (Welcome), 'Conocimientos y Experiencia' (Knowledge and Experience), 'Archivos Adjuntos' (Attachments), and 'Revisar y enviar' (Review and send), with the last step being the current one. An 'Enviar' (Send) button is located below the progress bar. The section is titled 'Revisar y enviar' (Review and send) in red. A message states: 'La siguiente información se enviará una vez que haga clic en el botón Enviar. Donde se muestra un enlace Editar, se puede modificar la información correspondiente.' (The following information will be sent once you click the Send button. Where a link Edit is shown, you can modify the corresponding information). Below this, there is a section for 'Bienvenid@' with an 'Editar' (Edit) link. The 'Información personal' (Personal information) section shows the name 'Raúl de la Torre Navarro'.

Figura 6. Look & Feel de información del candidato

Aquí vemos como visualiza su propia información el candidato, probablemente tampoco diste mucho de cómo ve la información detallada de cada uno de los candidatos el consultor encargado de hacer la selección.

Vemos claramente que el *software* parece cumplir con los requisitos para cumplir las necesidades de las empresas, sin embargo, como candidato se hace pesado de registrarse y es poco llamativo para la vista, pudiendo así aumentar el ratio de abandonos de la página.

2.2.5. Greenhouse

En la página principal de Greenhouse ("Recruiting Software Features | Greenhouse", 2017), encontraremos una página donde podremos ver todos los pasos de forma detallada.

Como observamos es una herramienta muy potente para realizar publicaciones en las distintas redes sociales y portales de empleo, consiguiendo así unos ratios de atracción muy buenos. Vemos que sigue la metodología típica en cuanto a arquitectura se refiere de todos los procesos de selección, proporcionándoles información muy detallada de los candidatos, pudiendo sacar unos informes de estos muy positivos para entregar a cliente.

Sin embargo, no podemos ver un control óptimo de todos los KPI's que podrían sernos útiles de cara a mejorar en futuros procesos, pudiendo, por ejemplo, sesgar mejor a que parte de la población va dirigida la oferta de empleo.

Tampoco vemos la posibilidad de filtrar por distintos factores que nos permite "rankear" a los candidatos de forma idónea".

2.2.6. NewtonSoftware

Sobre Newton ("Online Recruitment Management System Features | Newton Software", 2017), vemos que se trata de una herramienta muy completa para realizar los distintos tipos de gestión en un proceso de selección, pero por lo que podemos ver en sus distintos diseños web de cara al usuario que se registrará a una oferta de empleo le falta todavía mucho por mejorar. Probablemente esto provoque que la tasa de abandono de los candidatos de la página web antes de registrarse decrece, consiguiendo unos ratios peores de los esperados por muy buena y atractiva que sea la oferta de empleo.

2.2.7. Talent Clue

Toda la información posible sobre esta solución la encontramos en la página web de su empresa ("Talent Clue: software de reclutamiento para atraer y fidelizar", 2017).

Podemos apreciar que la solución de esta empresa es muy similar a la que utilizaremos nosotros. Sin embargo, vemos que tiene una fuerte carencia de control de los distintos KPI's que podrían resultar de utilidad para ayudar a mejorar en futuros procesos de selección, no teniendo los suficientes *widgets* con esta información relevante.

También apreciamos que hace uso de formularios ultra simple y responsivos, lo que nos hace pensar que al no mencionar nada sobre el *dashboard* que utilizarán los consultores este no será responsivo. Añadiremos además que un formulario sencillo puede proporcionar un mayor aumento de registro de candidatos, pero también el exceso de simplicidad puede provocar el efecto contrario, haciendo que el ratio de abandono en la página del formulario de inscripción aumente considerablemente.

Miramos a continuación, el *look&feel* de su aplicación y los distintos pasos que sigue para atraer talento, ya que a pesar de distintos puntos que consideramos podrían ser un problema, estamos ante un *software* que es bastante similar a la solución que nosotros queremos implantar.

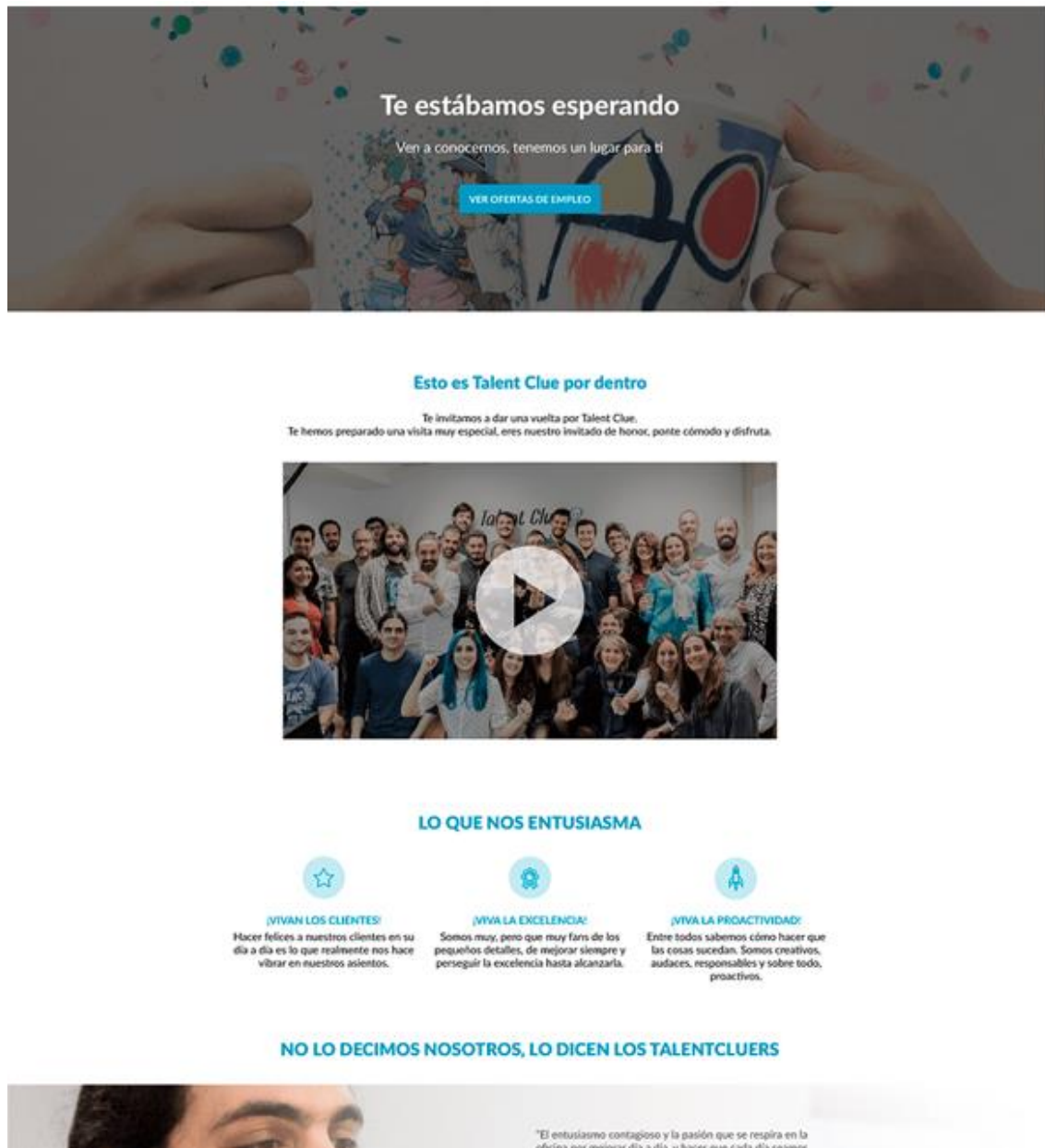


Figura 7. Look & Feel de Landing Page

Vemos como para atraer talento usan una página web responsiva y customizada al gusto del cliente, cumpliendo con su guía de marca, consiguiendo presencia en redes sociales.



Talent Clue

Talent Clue

DISEÑADOR/A GRÁFICO

Localización	Barcelona	Contrato	Prácticas
Jornada	Completa	Sector	Marketing
Vacantes	1	Sueldo	16.000€/año

QUIERO INSCRIBIRME

Quiero Ver Todas las Ofertas

¡Comparte esta Oferta!

f t in g+

TALENT CLUE SOLUTIONS S.L. Talent Clue es la empresa líder en creación y desarrollo de software para reclutadores que humaniza el proceso de selección. Somos más de 50 profesionales de varias especialidades trabajando para más de 450 clientes en más de 12 países. En Talent Clue apoyamos la igualdad, fomentamos la inclusión y celebramos la diversidad cada día, porque el talento de nuestros #TCHumans es lo que nos hará crecer.



DESCRIPCIÓN DE LA OFERTA

¿Estás buscando prácticas de diseño? En Talent Clue estamos buscando a un/a diseñador/a en prácticas para nuestro departamento de Marketing. Podrás vivir desde dentro cómo es estar en una start-up en pleno crecimiento, así que esperamos que tengas muchas muchísimas ganas de aprender :)

Te buscamos a ti, si criticas las cartas de los restaurantes a los que vas, si tienes miedo de que algún día te pasen un logo en un documento word (lo harán, te lo aseguro), si sabes adaptarte a los cambios y no lo ves todo en ##### y #000000... Pero sobretodo si odias la comic sans y jamás mandas un archivo a imprimir en RGB.

¿Sobre nosotros? En el equipo de Marketing somos 6 personas muy jóvenes con mucha ambición, locamente apasionados y enamorados de lo que hacemos. Nos movemos en el ámbito de España y Latinoamérica, y estamos creciendo. Pero para crecer más, te necesitamos.

¿De qué te encargarás?

- Maquetación de ebooks y presentaciones.
- Diseño de imágenes para redes sociales y CTAs para nuestro blog.
- Ayudar al equipo con el material gráfico que necesiten.

Si te ves diseñando con nosotros... ¡te estamos buscando y tú ya nos has encontrado!

Y en Talent Clue, a parte del equipo de Marketing, trabajarás en un equipo divertido, joven pero con las ideas claras, simpático y amante de los gatos (todos tenemos). Y también un poco "freak". La mayoría son programadores. Pero esto queda entre tú y yo. :)

¡Envíanos tu CV y asegúrate de incluir un link a tu portfolio!

REQUISITOS

Eres el/la candidato ideal para ser diseñador/a en Talent Clue:

- Si conoces Creative Cloud de Adobe y lo usas hasta para hacerte los horarios (sobretudo Photoshop, Illustrator e Indesign).
- Si eres organizado/a y sabes adaptarte a cambios e imprevistos.
- Si eres creativo/a y proactivo/a, de verdad de la buena.

Figura 8. Look & Feel de Oferta

Desde aquí el candidato podrá observar la oferta de trabajo e inscribirse a ella, no obstante, consideramos como experiencia de usuario y candidato que la descripción debería darse en la *Landing Page* de forma muy resumida para atraer el mayor número de posibles

candidatos posibles. Y una vez tengamos el talento en nuestra plataforma seleccionar al mejor y detallarle más los aspectos de la oferta.

2.2.8. Comparativa genérica de soluciones

En este sub apartado mostraremos una tabla haciendo especial hincapié en las ventajas y desventajas de todas las aplicaciones expuestas con anterioridad. Tendremos así una mejor comparación de las distintas aplicaciones.

	Ventajas	Desventajas
FileFinder	Gran cantidad de funcionalidades implementadas que da un gran potencial a la aplicación.	Cabe destacar la gran lentitud con la que trabaja la aplicación, así como su versión de escritorio y la necesidad de tener requisitos mínimos.
Zoho Recruit	Aplicación de coste muy bajo, recomendada para empresas muy pequeñas con escasos trabajadores.	Desarrollos caros de realizar, <i>look&feel</i> de la aplicación podría generar abandonos por parte de los candidatos en los procesos de selección, así como las funciones básicas como filtrado son bastante reducidas.
Bullhorn	Posibilidad de mostrar analíticas sobre los proyectos y mejorar en su organización.	Ausencia de realizar un filtrado optimo, al igual que un seguimiento adecuado de los candidatos inscritos.
Oracle Taleo Cloud Service	Aplicación como muchas de las funcionalidades deseadas en una buena solución de <i>software</i> ATS.	Aplicación poco atractiva de cara a los candidatos y a los usuarios que usen su plataforma.
Greenhouse	Información muy detallada de todo el proceso, así como dinamizando este en las distintas redes sociales, aumentando el ratio de inscritos.	Ausencia de posible control de KPI's.

NewtonSoftware	Solución muy completa, posee muchas funcionalidades.	Mucho por mejorar en sus tareas de <i>look&feel</i> .
Talent Clue	Similar a la aplicación que queremos implementar.	Formulario demasiado simple, posible aumento de abandono de potenciales candidatos.

Tabla 1. Comparativa del estado del arte

Como conclusión a las soluciones estudiadas, respecto a la que nosotros desarrollaremos a lo largo de este proyecto:

1. No son soluciones multidispositivo.
2. Algunas de ellas son soluciones de escritorio, es decir, previa instalación en el equipo.
3. *Look&feel* descuidado, puede provocar una pérdida considerable de potenciales candidatos.
4. No poseen control sobre los distintos KPI's del proyecto que puede extraerse de los datos que tengan.
5. No generan dinamismo en el flujo de trabajo.
6. Falta de proporcionar ahorro de tiempo que podría invertirse en la realización de otras tareas.

2.3. Tecnologías utilizadas

A continuación, detallaremos un poco las tecnologías que usaremos para el desarrollo de nuestra solución.

2.3.1. HTML5 y CSS3

“HTML5 es la última versión de HTML. El término representa dos conceptos diferentes:

- Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos.
- Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance. A este conjunto se le llama *HTML5 y amigos*, a menudo reducido a *HTML5*.” (“HTML5”, 2017)

Evidentemente, toda nuestra solución hará uso de HTML5 para ser soportado por todos los navegadores otorgándonos la capacidad de realizar toda función que necesitemos.

Así como el uso de CSS3, siendo esta la última versión de las hojas de estilo en cascada, trajo consigo una ampliación bastante considerable en comparación a CSS2, uso de gradientes, esquinas redondeadas, etc. El uso de estilos nos permitirá hacer que nuestra solución sea responsiva y pueda utilizarse en cualquier dispositivo de cualquier tamaño y resolución. será aplicable para todas las partes implicadas de nuestro desarrollo, poniendo especial hincapié en el uso de estilos, lo que verá el usuario que entra a registrarse a nuestra solución. ("CSS3", 2017)



Figura 9. Logo HTML5 y CSS3

2.3.2. JavaScript

“JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (*client-side*), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.” (“JavaScript”, 2017)

Utilizaremos JavaScript para hacer que nuestra *Landing Page* sea más dinámica y que los usuarios puedan interactuar con las redes sociales, así como conseguir las analíticas que nos deberá proporcionar del tráfico de datos de la página. Será útil a la hora de realizar un

formulario dinámico y que no resulte tedioso de cumplimentar para la inscripción de usuarios, realizando así una mejora considerable en la usabilidad de nuestro proyecto.



Figura 10. Logo de JavaScript

2.3.3. jQuery

“jQuery es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.” (“jQuery”, 2017)

Haremos uso de jQuery de la misma forma que usamos JavaScript, con el fin de mejorar la experiencia de usuario creando mínimas animaciones que mejoren la usabilidad de cara a nuestros futuros candidatos.



Figura 11. Logo de jQuery

2.3.4. Python

“Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License,¹ que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.” (“Python”, 2017)

El uso de Python será para realizar asignaciones a los candidatos que apliquen unas puntuaciones dependiendo de sus distintas respuestas. Podremos así realizar un *ranking* de los candidatos que facilitará en gran medida la tarea del consultor de RRHH.



Figura 12. Logo de Python

2.3.5. SQL

“SQL (por sus siglas en inglés Structured Query Language; en español lenguaje de consulta estructurada) es un lenguaje específico del dominio que da acceso a un sistema de gestión de bases de datos relacionales que permite especificar diversos tipos de operaciones en ellos. Una de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar, de forma sencilla, información de bases de datos, así como hacer cambios en ellas. SQL (Lenguaje estructurado de consulta) es un lenguaje específico de dominio que se usa en la programación y está diseñado para manejar los datos que hay en un sistema de base de datos, o para el procesamiento de flujos de datos en un sistema de gestión de flujos.

Originalmente está basado en el álgebra relacional y en el cálculo relacional, SQL consiste en un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de control de datos. El alcance de SQL incluye la inserción de datos, consultas, actualizaciones y borrado, la creación y modificación de esquemas y el control de acceso a los datos. También el SQL a veces se describe como un lenguaje declarativo, también incluye elementos procesales.

SQL fue uno de los primeros lenguajes comerciales para el modelo relacional de Edgar F. como se describió en su papel de 1970 “ El modelo relacional de datos para grandes bancos de datos compartidos”. A pesar de no adherirse totalmente al modelo relacional descrito por Codd, pasó a ser el lenguaje de base de datos más usado.

SQL pasó a ser el estándar del Instituto Americano Nacional de estándares en 1986 y de la Organización Internacional para estandarizar en 1987. Desde entonces. El estándar ha sido revisado para incluir más características. A pesar de la existencia de ambos estándares, La mayoría de los códigos SQL no son completamente portables entre sistemas de bases de datos diferentes sin ajustes.” ("SQL", 2017)

Este lenguaje será utilizado en nuestra aplicación para la comprobación de los datos respondidos por los candidatos, aplicando los criterios seleccionados más adelante para la selección inicial de estos, aplicando así las llamadas *killer questions*.



Figura 13. Logo de Oracle SQL Developer

2.3.6. JSON

“JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje, pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.” ("JSON", 2017)

Básicamente, JSON es un código de clave-valor, siendo esto en la mayoría de los lenguajes implementado como vectores, arreglos, listas, etc.



Figura 14. Logo JSON

2.3.7. SublimeText 3

“Sublime Text es un editor de texto y editor de código fuente está escrito en C++ y Python para los plugins. Desarrollado originalmente como una extensión de Vim, con el tiempo fue creando una identidad propia, por esto aún conserva un modo de edición tipo vi llamado *Vintage mode*.”

Se puede descargar y evaluar de forma gratuita. Sin embargo no es software libre o de código abierto³ y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad.” (“Sublime Text”, 2017)

Trabajaremos con Sublime Text para las tareas de programación ya que al ser un editor de texto fuertemente tipado nos permitirá optimizar nuestro flujo de trabajo. Podremos realizar tareas como tener un código en formato *pretty*, es decir, cumpliendo con normas de estilo, saltos de línea, etc. Estas acciones serán llevadas a cabo a través del uso del comando: Ctrl+Shift+P y usando la característica que deseemos.

Evidentemente, antes de poder realizar lo mencionado, con anterioridad deberemos tener los paquetes que nos proporcionarán estas funciones instalados. Usando Ctrl+Shift+P indicaremos *Install Package* y buscaremos el paquete que deseemos.

Por ejemplo, nos sentiremos cómodos trabajando con los siguientes paquetes:

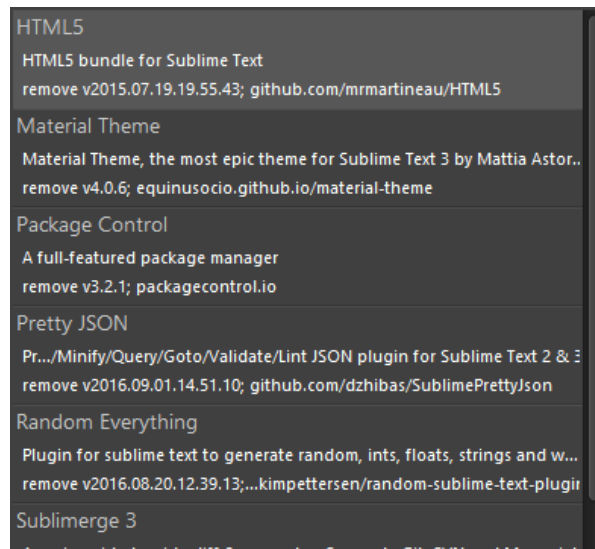


Figura 15. Ejemplo de paquetes instalados

Siendo de gran utilidad Sublimerge 3 que nos permitirá comparar un código con otro copiado al portapapeles, para poder realizar las comprobaciones oportunas sobre que hemos modificado respecto a lo anterior y depurar fallos de una forma relativamente más eficiente que a mano.



Figura 16. Logo de Sublime Text 3

2.3.8. Trello

“Trello es una aplicación web (está en la nube) para hacer listas dentro de listas (describiéndola de un modo muy básico). Lo que la hace única, es su versatilidad: A cada elemento de una lista se le puede agregar de todo: otras listas, imágenes, vídeos, documentos, etc. Además, es extremadamente potente para uso colaborativo. Pueden agregarse cualquier número de usuarios, asignar tareas, ponerles fecha límite (tiene su propio calendario, que puedes sincronizar con el de Google), etc.

Se basa en el método Kanban para gestión de proyectos, con tarjetas que viajan por diferentes listas en función de su estado: Así, solemos tener una lista de cosas por hacer (to do, o pendientes), que se están haciendo (doing, o en proceso) o hechas

(done, o terminadas)." ("Trello: el administrador de proyectos definitivo - Dispersium", 2017)

Trello evidentemente es una aplicación completamente gratuita, no obstante, si pagamos podremos recibir algunas funcionalidades extra, pero para la organización de nuestras tareas y la de nuestro equipo de trabajo no será necesario.

A través de esta herramienta podremos observar el progreso de nuestro proyecto de forma telemática, siendo posible el ver el estado de las tareas de cada uno de los miembros del equipo. Así podremos saber que tareas ha terminado, cuales son las que está realizando y las que todavía tiene encoladas en su lista de tareas por hacer.

Todo esto nos permitirá manejar tiempos con mayor soltura y estar pendientes del estado de trabajo de nuestros compañeros, sin necesidad de tener reuniones periódicas para ver el estado en el que se encuentra cada uno de los miembros del equipo en la realización de sus tareas.

En el siguiente enlace podremos ver un breve tutorial de lo que es Trello ("Trello - organización on-line y cooperativa. Tutorial subtítulo Español", 2017) y a continuación tendremos una guía escrita sobre el gestor de tareas Trello con Kanban (Torres, Torres, & Torres, 2017).

En nuestro caso, utilizamos Trello para organizar el proyecto de una forma más eficiente con el tutor del Trabajo Fin de Grado, lo que nos permite a los dos estar en contacto y ver un progreso de una forma más dinámica y eficiente, además de tener reuniones semanales donde detallar y pulir distintos aspectos del proyecto.

En la siguiente imagen veremos el kanban que hemos usado para organizar nuestro proyecto:

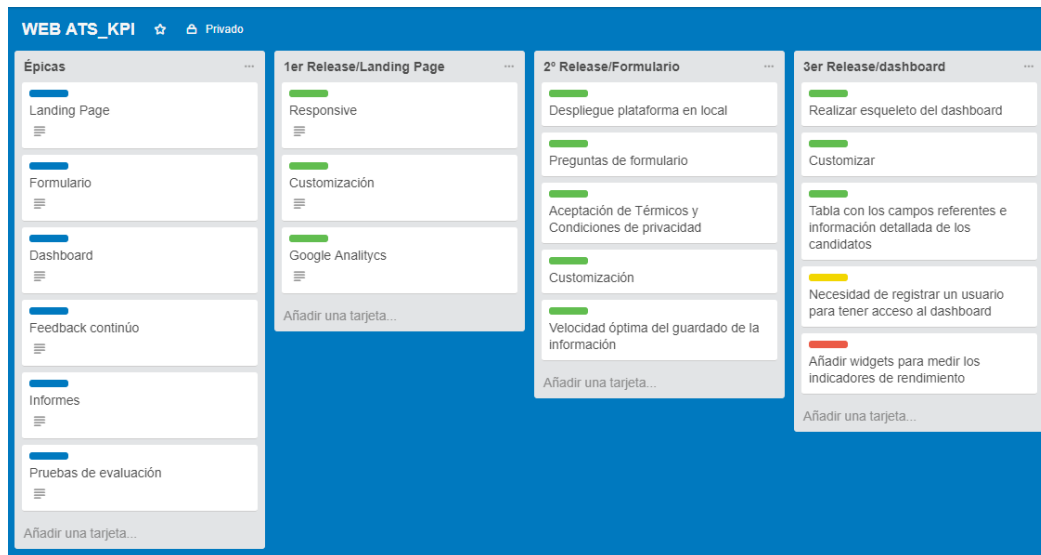


Figura 17. Uso de Trello

Este kanban se ha usado para una implementación más detallada de nuestra aplicación. Siendo la organización de dicho kanban la que describimos a continuación: a la izquierda del todo tendremos una descripción detallada de las épicas de nuestro proyecto. Después dividimos dicho proyecto en distintas entregas o versiones, dividiendo éstas entregas por tareas. Las tareas completadas aparecerán en verde, las que se están realizando en amarillo y las que todavía no se han comenzado en rojo. Cuanto más arriba estén en la pila mayor será la importancia de dichas actividades.



Figura 18. Logo de Trello

Añadir que también tenemos la opción de descargarnos la aplicación para nuestro móvil.

Capítulo 3. Metodología de Trabajo (Scrum)

En este capítulo hablaremos sobre la metodología que vamos a utilizar para el desarrollo de nuestra plataforma web, indagaremos un poco más a fondo en esta metodología en los siguientes puntos.

3.1. Historia

“El concepto de Scrum tiene su origen en un estudio de 1986 sobre los nuevos procesos de desarrollo utilizados en productos exitosos en Japón y los Estados Unidos (cámaras de fotos de Canon, fotocopadoras de Xerox, automóviles de Honda, ordenadores de HP y otros). Los equipos que desarrollaron estos productos partían de requisitos muy generales, así como novedosos, y debían salir al mercado en mucho menos del tiempo del que se tardó en lanzar productos anteriores. Estos equipos seguían patrones de ejecución de proyecto muy similares. En este estudio se comparaba la forma de trabajo de estos equipos altamente productivos y multidisciplinarios con la colaboración entre los jugadores de Rugby y su formación de Scrum (melé en español).

En 1993 se realizó el primer Scrum para desarrollo de software y en 1995 el proceso fue formalizado. En 2001 un grupo de personas muy relevantes en lo que empezaba a ser el desarrollo ágil escribieron los valores fundamentales de los procesos ágiles.

Desde 1995 miles de proyectos en todo el mundo han utilizado Scrum para el desarrollo de productos, tanto en empresas pequeñas, “startups” con tan sólo 3 personas desarrollando un producto, como en multinacionales.” (“Historia de Scrum”, 2017)

3.2. Introducción

Describimos la implementación de la metodología de trabajo scrum en la institución Universidad Carlos III de Madrid para la gestión del desarrollo del proyecto Aplicación Web para el seguimiento de Candidatos y Control de Indicadores Claves de Rendimientos en Procesos de Selección de Recursos Humanos. Incluiremos junto con esta descripción el ciclo de vida iterativo e incremental para el proyecto, los artefactos o documentos con los que se

gestionan las tareas de adquisición y suministro: requisitos, monitorización y seguimiento del avance, así como las responsabilidades y compromisos de los participantes en el proyecto.

3.3. Descripción General de la Metodología

3.3.1. Fundamentación

Las principales razones del uso de un ciclo de desarrollo iterativo e incremental de tipo scrum para la ejecución de dicho proyecto son:

1. **Sistema modular:** Las características del sistema Aplicación Web para el seguimiento de Candidatos y Control de Indicadores Claves de Rendimientos en Procesos de Selección de Recursos Humanos permiten desarrollar una base funcional mínima y sobre ella ir incrementando las funcionalidades o modificando el comportamiento o apariencia de las ya implementadas.
2. **Entregas frecuentes y continuas al cliente:** dichas entregas se realizarán con los módulos correspondientes terminados, pudiendo así disponer de una funcionalidad básica en un tiempo mínimo y a partir de ahí un incremento y mejora continua del sistema.
3. **Previsible inestabilidad de requisitos:**
 - a. Es muy probable que el sistema incorpore más funcionalidades de las inicialmente identificadas.
 - b. Es posible que durante la ejecución del proyecto se altere el orden en el que se desean recibir módulos o historias de usuario terminadas
 - c. Para el cliente resulta difícil precisar cuál será la dimensión completa del sistema, y su crecimiento puede continuarse en el tiempo, suspenderse o detenerse.

3.3.2. Valores de trabajo

Los valores que deben ser practicados por todos los miembros involucrados en el desarrollo y que hacen posible que la metodología Scrum tenga éxito son:

1. Autonomía del equipo
2. Respeto en el equipo

3. Responsabilidad y auto-disciplina
4. Foco en la tarea
5. Información, transparencia y visibilidad

3.3.3. Roles del proyecto

Encontraremos varios tipos de roles, entre los que destacaremos los siguientes:

- Cliente
- Scrum Manager / Coordinador
- Gestor de Producto / Product Owner
- Equipo técnico

3.4. Requisitos de nuestro sistema

En este pequeño epígrafe hablaremos sobre las épicas, que no son más que algo que está por encima de las historias de usuario (de estas hablaremos más adelante en el análisis del sistema), teniendo así una idea abstracta de lo que se quiere conseguir con las épicas. Podrán ser realizadas en varias interacciones. Veremos más adelante que para completar alguna épica necesitaremos seguir avanzando en el desarrollo de nuestra aplicación. Añadiendo poco a poco las funcionalidades que completen dicha épica.

1) *Landing Page*



Figura 19. Épica (I)

Necesitaremos como bien se comenta en la épica una *Landing Page* que podamos usar a modo de plantilla para otros proyectos. Teniendo esta plantilla podremos optimizar su explotación, teniendo única y exclusivamente que cambiar sus colores, logos, etc.

2) Formulario

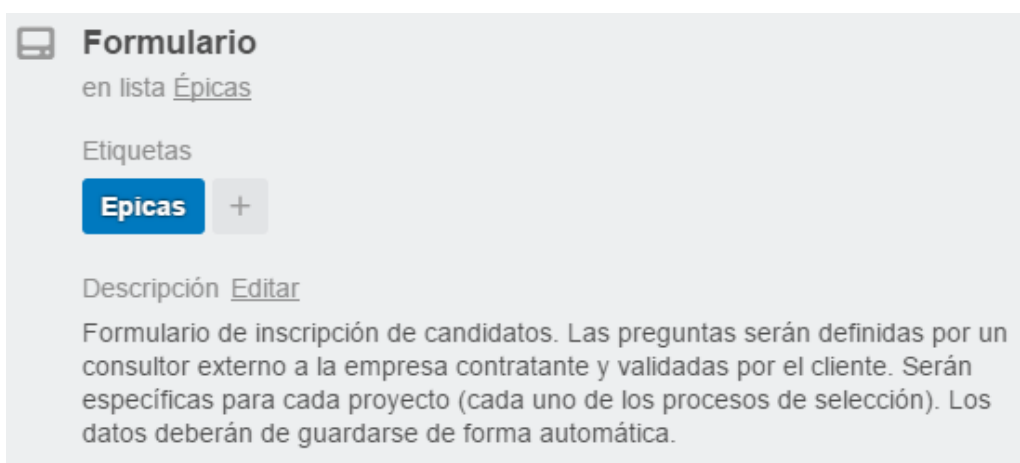


Figura 20. Epica (II)

El formulario también deberá de seguir la guía de marca de la respectiva compañía. Pudiendo cambiar sus preguntas y respuestas, pero siguiendo siempre el mismo *Look&Feel*.

Cualquier equipo como: un PC de sobremesa, un portátil o un Smartphone. Cualquiera de estos podrá tener cualquier tipo de Sistema Operativo instalado y nuestra solución funcionará siempre y cuando disponga de conexión a internet y de un navegador.

3) *Dashboard*



Figura 21. Épica (III)

Deberemos estudiar más adelante como realizaremos los distintos *widgets* que queremos usar para el *dashboard* y que nos puedan aportar información de valor, facilitando así las tareas del consultor de RRHH.

4) *Feedback* continuo

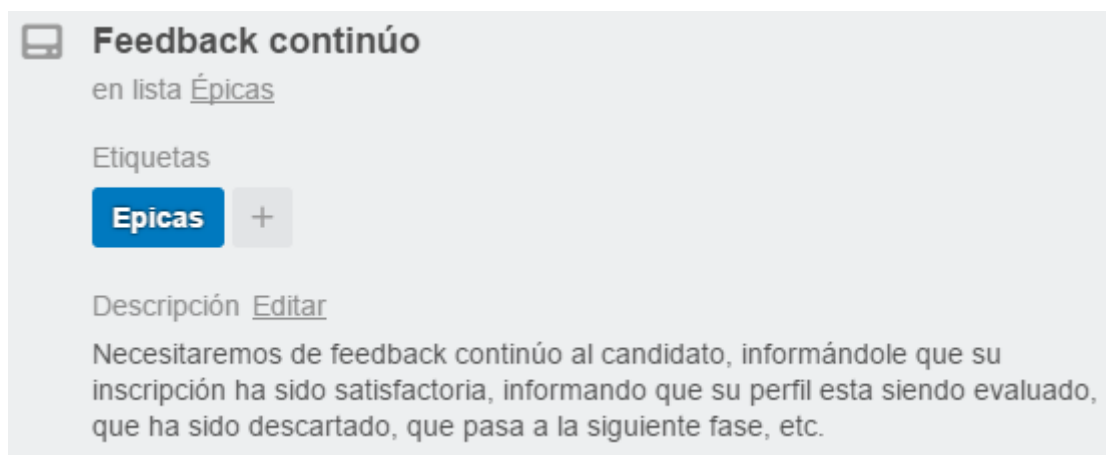


Figura 22. Épica (IV)

Será requisito indispensable el continuo *feedback* a los candidatos de su situación en el proceso, para ello haremos un *template* de correo listo para cambiar únicamente los textos a gusto del cliente, así como su logo. Deberán de ser correos multidispositivo, teniendo en

cuenta también la posibilidad de que sean visualizados en distintas aplicaciones como: Gmail, Outlook, Thunderbird, etc.

5) Informes

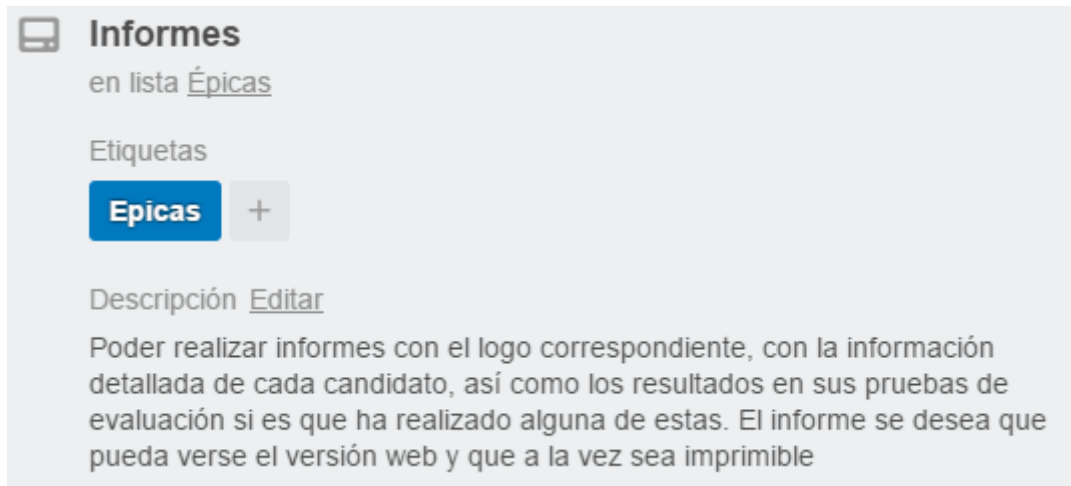


Figura 23. Épica (V)

Ver la forma de implementar el poder hacer informes individuales para cada candidato, también sería deseable poder modificar su información en el informe sin modificar la información extraída de lo que aporte el candidato a la hora de inscribirse.

6) Evaluación del candidato

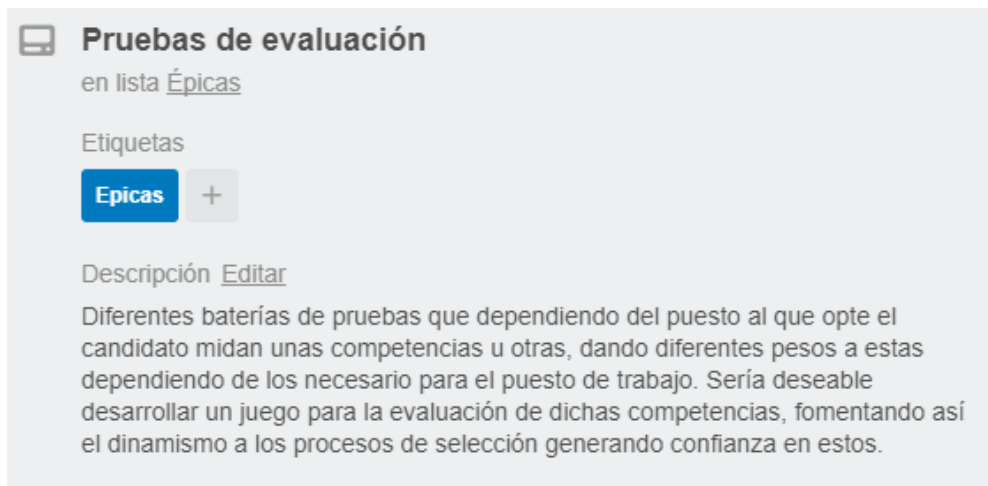


Figura 24. Épica (VI)

Se desarrollará un juego que haga posible el medir las competencias de dichos candidatos, también otra opción posible será buscar una empresa que proporcione estos servicios y ver la manera de implementar este juego en nuestro proceso de selección.

Capítulo 4. Análisis del Sistema

Haremos uso de metodologías ágiles en el desarrollo de nuestro *software*, así, lo deseado por nosotros será alcanzar hasta el último objetivo, sin embargo, siendo realistas y por cuestión de tiempos empleados en la realización del desarrollo y la memoria podría darse el caso en que nos encontrásemos en un *release* que no cumpliera con todo lo anterior. Lo detallaremos todo más adelante según vayamos cumpliendo con las distintas entregas, las cuales también detallaremos más adelante.

En este capítulo haremos referencia en forma de historias de usuario, a las capacidades, funcionalidades y restricciones de los sistemas.

Cubriremos todas las necesidades que un consultor de RRHH necesita para que los potenciales candidatos vayan a inscribirse en la oferta de empleo, ayudando al consultor a realizar su trabajo de una forma más eficiente.

La organización del proyecto será a través de Trello, por sencillez y rapidez.

Además, en este epígrafe, explicaremos todos los puntos de relevancia sobre la metodología utilizada, el plan de producto y el gráfico de *Burn Down* que los incluiremos en dicho capítulo, el primero en el *product backlog* para la organización de las tareas a realizar en cada uno de los *sprint*, dando así mayor detalle después al *sprint backlog*, el gráfico *burn down* estará relacionado con el actual progreso del *sprint* en curso actualmente. ("ScrumManager - Inicio", 2017)

A continuación, a modo de simulación asignaremos los roles de un equipo de trabajo, para tener un posible ejemplo real.

1. Product Owner: el departamento de Recursos Humanos
2. Scrum manager: en este particular caso podría tratarse del tutor José María Rubio Manso.
3. Equipo técnico: integrado idealmente por cuatro personas, con mayor o menor experiencia en las tecnologías utilizadas para el desarrollo de la plataforma.
4. Resto de implicados: la empresa que contrata nuestros servicios.

Siendo estos un ejemplo de roles a tener en cuenta en un proyecto real, en el caso de querer seguir adelante con nuestra solución tecnológica.

No obstante, todos los roles de dicho proyecto han recaído en este particular caso sobre el autor de dicho proyecto, Raúl de la Torre Navarro. Lo anteriormente mencionado es de cara a una posible simulación en el mundo profesional. La realidad en este caso ha sido que todos los roles han sido adquiridos por una única persona siendo esto positivo de cara a ver todos los puntos de vista implicados desde distintas perspectivas, dando un abanico más amplio a la visión de la realización de proyectos.

4.1. Product Backlog

En este epígrafe hablaremos sobre el plan de producto, así como sus distintas entregas. Añadir que cada entrega/*release* coincidirá con un *sprint* de nuestra aplicación, no tiene porque suceder así, pero en este caso se tomó dicha decisión al comienzo del proyecto. Será nombrado varias veces a lo largo del proyecto, debido a que es un punto muy importante a considerar por ser parte de nuestras entregas.

4.1.1. Plan de producto

Como bien indica su nombre, es una representación del plan de producto previsto por el gestor de producto. Representa las tareas del sistema en el orden en que se desean, así como su tiempo de ejecución prevista.

4.1.1.1. Responsabilidades de los implicados

- Responsabilidad del gestor de producto:
 1. Confección del proyecto
 2. Mantenimiento actualizado en todo momento durante la ejecución del proyecto
 - a) Orden en el que se desea disponer de los temas o *epics* del sistema e hitos del producto (versiones).
 - b) Incorporación, eliminación, modificaciones de los temas, de su orden de prioridad, estimaciones o hitos.
- Responsabilidades del Scrum Manager:

1. Supervisión del gráfico de producto y comunicación con el gestor de producto para pedir aclaración de dudas que pueda tener, así como asesoramiento para la subsanación de las deficiencias observadas.
- Responsabilidades del equipo técnico:
 1. Conocimiento y comprensión actualizada del plan de producto.
 2. Resolución de dudas o comunicación de sugerencias con Scrum Manager o gestor de producto. Dependerá según se haya implantado en la empresa.
 - Responsabilidades del resto de implicados:
 1. Conocimiento y comprensión actualizada del plan de producto.
 2. Resolución de dudas o comunicación de sugerencias con Scrum Manager o gestor de producto. Dependerá según se haya implantado en la empresa.

A continuación, mostramos la tabla que representará el plan de producto:

ID	Tarea	Trabajo (horas)	Criterio de validación
1	<i>Responsive</i>	14	Realizar pruebas pertinentes
2	Customización	20	Realizar pruebas pertinentes
3	Google Analytics	6	Realizar pruebas pertinentes
3.1	Hito (1er <i>release</i>)	Total: 40	Entrega de incremento
4	Despliegue en local	20	Realizar pruebas pertinentes
5	Preguntas formulario	25	Realizar pruebas pertinentes
6	Términos y condiciones	20	Realizar pruebas pertinentes
7	Customización formulario	13	Realizar pruebas pertinentes
8	Testeo velocidad óptima de guardado	2	Realizar pruebas pertinentes
8.1	Hito (2º <i>release</i>)	Total: 80	Entrega de incremento
9	Esqueleto <i>dashboard</i>	20	Realizar pruebas pertinentes
10	Customización <i>dashboard</i>	5	Realizar pruebas pertinentes
11	Tabla con información	35	Realizar pruebas pertinentes

12	Registro de usuarios con acceso al <i>dashboard</i>	20	Realizar pruebas pertinentes
13	<i>Widgets</i> para medición de KPI's	40	Realizar pruebas pertinentes
13.1	Hito (3er <i>release</i>)	Total: 120	Entrega de incremento
14	Introducción de estados	Por determinar	Realizar pruebas pertinentes
15	Condiciones de cambio de estado. Cribas curriculares	Por determinar	Realizar pruebas pertinentes
16	Maquetado de correos de <i>feedback</i> enviados desde la plataforma	Por determinar	Realizar pruebas pertinentes
16.1	Hito (4º <i>release</i>)	Por determinar	Entrega de incremento
17	Maquetar en html un esqueleto para informes. Responsivo	Por determinar	Realizar pruebas pertinentes
18	Realizar la posibilidad de imprimir en formato papel	Por determinar	Realizar pruebas pertinentes
19	Estudiar la mejor forma de mostrar toda la información oportuna	Por determinar	Realizar pruebas pertinentes
19.1	Hito (5º <i>release</i>)	Por determinar	Entrega de incremento
20	Análisis de videojuego para determinar perfil y conocimientos	Por determinar	Entrega de incremento
21	Diseño de	Por determinar	Entrega de incremento

	videojuego para determinar perfil y conocimientos		
22	Juego completamente multidispositivo	Por determinar	Entrega de incremento
22.1	Hito (6ª release)	Por determinar	Entrega de incremento

Tabla 2. Plan de Producto

En este caso hemos añadido a la planificación de proyecto: los incrementos ya entregados, el tercer *release* que es en el que estamos ocupados actualmente y el cuarto, quinto y sexto incremento que serán las siguientes entregas de nuestro producto, los cuales realizaremos una breve estimación. Esta estimación podemos realizarla debido a que ya conocemos el tiempo de trabajo requerido aproximado para cada tipo de tarea.

Iremos así planificado de forma modular el proyecto según vayamos avanzando y tengamos bien especificados los pasos a dar de forma clara en el proyecto. Teniendo así estimaciones más exactas.

4.2.1. Responsabilidades de los implicados en el Product Backlog

- Gestor de producto/Product Owner:
 1. Registro en la lista de la pila del producto de las historias de usuario que definen el sistema.
 2. Mantenimiento actualizado de la pila de producto en todo momento durante la ejecución del proyecto
 - a. Orden deseado en el que quiere recibir terminada cada historia de usuario.
 - b. Incorporación/eliminación/modificación de las historias o de su orden de prioridad.
 - c. Disponibilidad: Manteniendo la comunicación a través de la herramienta de Trello, enviando las modificaciones al Scrum Manager para su actualización.
- Responsabilidades del Scrum Manager/Coordinador:

1. Supervisión de la pila de producto, comunicación con el gestor del producto para pedir aclaraciones de las dudas que puedan surgir, así como asesorar para la subsanación de las deficiencias observadas.
- Responsabilidades del equipo técnico:
 1. Conocimiento y comprensión de la pila de producto
 2. Resolución de dudas o comunicación de sugerencias con el gestor de producto, también podría darse la comunicación con el Scrum Manager o algún otro implicado, habría que indicarlo al principio del proyecto.
 - Responsabilidades del resto de implicados:
 1. Conocimiento y comprensión de la pila de producto
 2. Resolución de dudas o comunicación de sugerencias con el gestor de producto, también podría darse la comunicación con el Scrum Manager o algún otro implicado, habría que indicarlo al principio del proyecto.

El gestor de producto podrá solicitar asesoría al Scrum Manager del proyecto o al personal técnico del equipo para conocer la estimación temprana de las historias de usuario cuyo tamaño aproximado le presenten dudas.

4.2.2. Planificación Sprint

En este apartado hablaremos brevemente sobre la planificación de los *sprint* realizados y los que tenemos intención de realizar. Hablaremos también de una cifra en horas/puntos de los *sprint* ya realizados y el que todavía estamos realizando.

Si bien es cierto que los *sprint* siguen una línea similar a las épicas, las épicas no estarán completamente completadas hasta que vayamos realizando también futuras entregas, añadiendo más funcionalidades.

1. Primer *sprint*, constará de 40 horas de trabajo para la realización de la *Landing Page*. Estos tiempos se verán reducidos en futuras entregas, debido a la reutilización de códigos y la ganancia de experiencia.

2. Segundo *sprint*, donde contaremos con 80 horas de trabajo para la realización de la parte de formulario, empezando a dar funcionalidades a nuestra aplicación.
3. El tercer *sprint* está planificado para 120 horas de trabajo siendo en este *sprint* la principal tarea la de realizar un *dashboard* funcional. Este irá completándose con futuros *sprint*, completando así también en varias iteraciones la épica que podría corresponder a este.

A partir del tercer *sprint* ya no tenemos una valoración objetiva de las horas de duración de los distintos *sprint*, debido a la posibilidad de que haya cambios en los objetivos y tareas a realizar en estos *sprint* en el futuro. Sin embargo, podemos realizar una estimación de la duración total de los siguientes *release*.

4. El cuarto *sprint*, será la capacidad de dar *feedback* continuado a los candidatos, esto se realizará dotando de estados a los candidatos, estados automáticos que dependerán de las condiciones acordadas y lo respondido por los candidatos en el formulario. Como bien podemos observar, este *sprint* irá ligado también al de formulario, dotando a este de nuevas funcionalidades a nuestra aplicación, completando un poco más dicha épica. Estimamos el tiempo de ejecución de este *sprint* en 80 horas.
5. En el quinto *sprint* dotaremos de la capacidad de realizar informes individuales desde el *dashboard*. Al igual que en el cuarto *sprint*, este ayudará también a completar la tercera épica. Este *sprint* su estimación en horas será de 120.
6. Sexto *sprint*, implantación de pruebas de evaluación por parte de los candidatos. Ya sean pruebas típicas o a través de la realización de un juego. Este *sprint* llevará adjudicado una estimación de 80 horas para su realización.

4.3. Sprint Backlog

Describiremos el registro de los requisitos detallados o tareas que va a desarrollar el equipo técnico, iremos dividiéndolo en distintos *releases* según vayamos avanzando en el desarrollo del proyecto.

Hemos completado un total de dos *release*, teniendo el tercero casi completado a falta de un par de tareas por implementar.

Veremos en cada uno de los *release* de los que hablamos, sus historias de usuario, planificando así todo de forma óptima.

La planificación de los distintos *releases* que expondremos a continuación coincidirán con las versiones de nuestro proyecto. No obstante, las distintas versiones que se irán desplegando irán teniendo efecto sobre las anteriores versiones, debido a que irá dotando al resto de más funcionalidades. Aportando así un mayor valor en cada versión realizada.

Veremos como hemos dividido las historias de usuario en los distintos *sprint/release*.

4.3.1. Responsabilidades de los implicados

- Gestor de producto/Product Owner:
 1. Presencia en las reuniones en las que se elabora la pila del *sprint*.
 2. Resolución de dudas sobre las historias de usuario en las que se descompone la pila del *sprint*.
- Responsabilidades del Scrum Manager/Coordinador:
 1. Supervisión y asesoría en la elaboración de la pila de del *sprint*.
- Responsabilidades del equipo técnico:
 1. Elaboración de la pila del *sprint*.
 2. Resolución de dudas o comunicación de posibles sugerencias sobre las historias de usuario o épicas con el gestor de producto.

4.3.2. Primer Release/Sprint

Enumeramos, a continuación, las historias de usuario del primer *release* que definirán el comportamiento de nuestra aplicación. Valoraremos el tipo y prioridad de estos.

Podemos leer a continuación una breve leyenda de las tareas importantes a realizar en este primer *Release*:

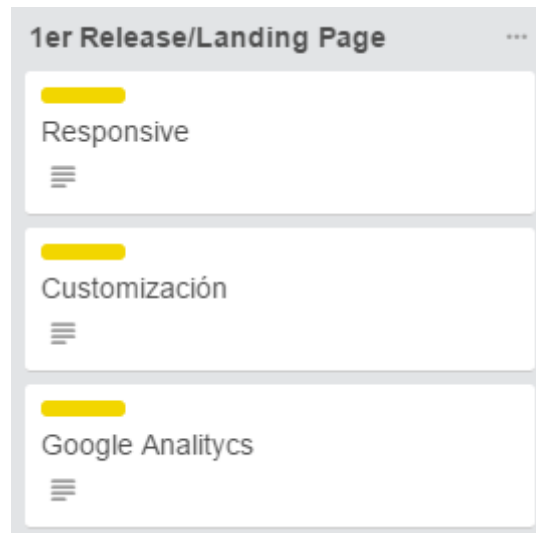


Figura 25. Tareas primer release

Las tareas están organizadas por orden de prioridad. Como vemos lo más prioritario es construir una página web que sea completamente multidispositivo, seguido de las siguientes tareas. Este *release* está previsto su realización en una semana de trabajo, 40 horas. Programación, textos, customización siguiendo la guía de marca del cliente así como testear que las visitas recibidas a la web aumentan de forma adecuada y se comparte el proceso de forma óptima en redes sociales.

1. Hacer uso del *framework bootstrap* y partir de plantillas gratuitas disponibles en internet.
2. Modificar dichas plantillas por si diese algún tipo de error cuando nos encontramos ante distintas dimensiones de pantalla.
3. Seguir la guía de marca del cliente, en caso de no ser proporcionada una guía de marca seguir las pautas señaladas por el cliente.
4. Introducir el código oportuno de Google Analytics y crear su propiedad.

Deberá ser un diseño intuitivo y acorde a la actualidad, evitando así posibles rebotes de potenciales candidatos.

Al terminar el primer *Sprint* deberemos tener la *Landing Page* lista para entregar. Añadir que las 40 horas semanales solo se cumplirán en el primer proyecto de todos. Una vez esta tarea tenga definido su *template* y esté preparada para simplemente cambiar logos, colores y textos deberá ser una tarea a realizar en 8 horas.

Vemos ahora un ejemplo de cómo sería una plantilla que representaría una historia de usuario.

Historia de Usuario	
ID:	Usuario:
Nombre historia:	
Prioridad en Negocio:	Riesgo en desarrollo:
Puntos estimados:	Iteración asignada:
Programador responsable: Raúl de la Torre Navarro	
Descripción:	
Validación:	

Tabla 3. Plantilla Historia de Usuario

Historia de Usuario	
ID: 1	Usuario: Cliente
Nombre historia: Responsividad de la página	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 15	Iteración asignada: 1
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea una página web completamente responsiva, poder visualizar dicha página desde cualquier dispositivo. Pudiendo acceder desde el ordenador, Tablet, Smartphone o cualquier otro dispositivo con capacidad para conectarse a internet, pudiendo acceder a la información desde cualquier punto.	
Validación: El cliente podrá utilizar la página web desde cualquier dispositivo.	

Tabla 4. Historia de Usuario (I)

Historia de Usuario	
ID: 2	Usuario: Cliente
Nombre historia: Customización de la página	
Prioridad en Negocio: Media	Riesgo en desarrollo: Bajo
Puntos estimados: 20	Iteración asignada: 2
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea que la página sea realizada siguiendo la guía de marca de su empresa, utilizando su paleta de colores, su logo, textos específicos para su solución. También desea poder interactuar con las distintas RRSS.	
Validación: El cliente podrá ver que la página sigue su guía de marca.	

Tabla 5. Historia de Usuario (II)

Historia de Usuario	
ID: 3	Usuario: Cliente
Nombre historia: Google Analytics	
Prioridad en Negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 5	Iteración asignada: 3
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario quiere poder realizar todas las métricas que estén a su alcance para tener un mayor control del proceso, pudiendo aprender de las métricas obtenidas y mejorar su sitio web para el futuro. Pudiendo ver también cuanto interactúa la gente con las RRSS, así como el tiempo de permanencia de las personas en el sitio web.	
Validación: El cliente podrá pedir informes sobre los datos obtenidos a través de google analytics.	

Tabla 6. Historia de Usuario (III)

4.3.2.1. Incremento

En este incremento hacemos entrega de la *Landing Page* acordada con el cliente con todo lo necesario para informar a los potenciales candidatos de nuestro cliente sobre el proceso al que van a aplicar.

Nos vimos obligados a realizar una pequeña tarea para el cumplimiento de las leyes y normativas vigentes en la actualidad:

Al introducir código de Google Analytics nos vemos obligados a hacer un aviso a los usuarios del uso de cookies ya que podríamos ser sancionados si no avisamos de ello. Así como referencia a un aviso legal, este dependerá de si los datos serán alojados en nuestras bases de datos o en las bases de datos de los clientes, dependiendo de lo acordado con estos. (Luna, 2017).

Respecto a la planificación del *product backlog* esta tarea extra a realizar no aporta mayor inconveniente, debido a que fue de rápida solución al encontrar muchos casos de forma cotidiana cuando navegamos por internet, pudiendo tomar ejemplo de estos, sin necesidad de realizar una investigación.

Comentar que de haber influido en nuestra planificación, gracias al uso de metodologías ágiles, podríamos haber modificado la pila de producto realizando alguna tarea

en el siguiente *release* o posponiéndose en el tiempo dicha tarea incluyéndola en futuras entregas. Priorizando siempre las tareas más importantes. Esta capacidad de modificar nuestro proyecto tan versátilmente supone un valor añadido a nuestro producto.



Figura 26. Figura de aviso de cookies

Así dejamos constancia del uso de cookies, pudiendo obtener más información sobre lo referente a la LOPD para no tener posibles sanciones debido a su incumplimiento.

4.3.3. Segundo Release/Sprint

Enumeramos las historias de usuario del segundo *release* que definirán el comportamiento de nuestra aplicación. Valoraremos el tipo y prioridad de estos.

Podemos leer a continuación una breve leyenda de las tareas importantes a realizar en este segundo *release*:

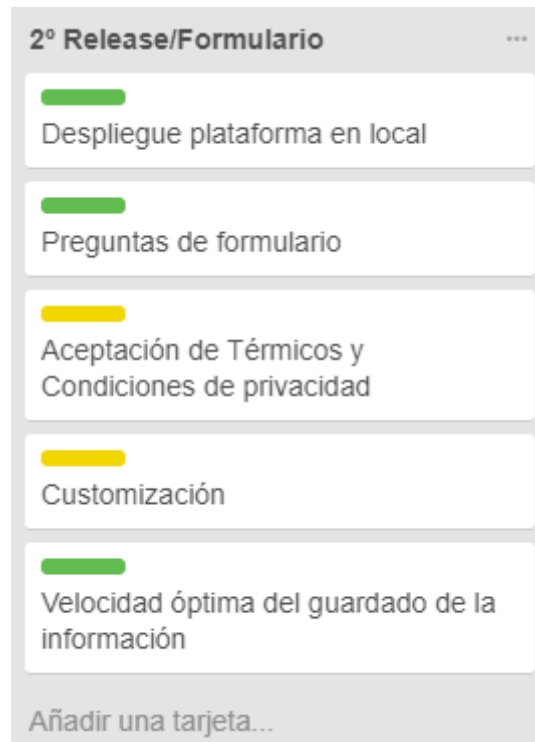


Figura 27. Tareas segundo release

Las tareas están organizadas por orden de prioridad. Como vemos lo más prioritario es evidentemente el despliegue de nuestra plataforma en local, seguido de las siguientes tareas. Este *release* está previsto su realización en dos semanas de trabajo, 80 horas. Programación, textos, customización siguiendo la guía de marca del cliente, así como testear que la velocidad de guardado de los datos de nuestro formulario es óptima.

1. Hacer uso del *framework bootstrap* y partir de plantillas gratuitas disponibles en internet, añadir que el formulario seguirá un estilo similar al de la *Landing Page*.
2. Modificar los estilos del formulario para conseguir una vista de estos moderna y dinámica.
3. Introducir un *checkbox* para la aceptación de términos y condiciones de seguridad, dicho texto dependerá de si la solución se encuentra alojada en nuestros servidores y somos los propietarios nosotros o nuestros clientes.
4. Testear la velocidad del envío de la información de nuestro formulario.

Deberá ser un diseño intuitivo y acorde a la actualidad, evitando así posibles rebotes de potenciales candidatos.

Al terminar el segundo *Sprint* deberemos tener el formulario listo para desplegar y hacerlo visible a los potenciales candidatos, dándoles opción a registrarse. En el caso de este *release*, podrán bajarse los tiempos de su realización una vez optimicemos el proceso, realizando una automatización de este. Es decir, la idea es poder tener un formulario realizando un simple json, desde una clase construida para el *site* del *admin* en Django, consiguiendo reducir enormemente los tiempos de programación. Esto será introducido como posibles mejoras en el futuro.

Historia de Usuario	
ID: 4	Usuario: Cliente
Nombre historia: Despliegue plataforma en local	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 22	Iteración asignada: 4
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea que antes de entregar cualquier versión, como es lógico, realicemos las pruebas oportunas. Por ello desea que desplaguemos de forma local para realizar pruebas reales.	
Validación: El cliente desea poder probar primeramente la aplicación en local para que realicemos las pruebas oportunas.	

Tabla 7. Historia de Usuario (IV)

Historia de Usuario	
ID: 5	Usuario: Cliente
Nombre historia: Preguntas	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 20	Iteración asignada: 5
Programador responsable: Raúl de la Torre Navarro	
Descripción: Preguntas que llevará el formulario, para obtener información relevante acerca del puesto al que optará el candidato. Las preguntas desea el usuario que sean validadas previamente por él, para después poder sacar los KPI's oportunos sobre las preguntas realizadas.	
Validación: El cliente validará con los responsable las preguntas a implementar en el formulario, pudiendo hacer cambios mínimos sobre estas.	

Tabla 8. Historia de Usuario (V)

Historia de Usuario	
ID: 6	Usuario: Cliente
Nombre historia: Términos y condiciones	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 20	Iteración asignada: 6
Programador responsable: Raúl de la Torre Navarro	
Descripción: Necesidad de indicar un campo de aceptación de términos y condiciones para aplicar a la oferta. Cumpliendo así con la LOPD vigente y evitar posibles problemas legales.	
Validación: El cliente validará junto con nuestro departamento legal si los términos y condiciones son los adecuados para la oferta y la empresa.	

Tabla 9. Historia de Usuario (VI)

Historia de Usuario	
ID: 7	Usuario: Cliente
Nombre historia: Customización	
Prioridad en Negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 13	Iteración asignada: 7
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea que la página sea realizada siguiendo la guía de marca de su empresa, utilizando su paleta de colores, su logo, textos específicos para su solución. También desea poder interactuar con las distintas RRSS.	
Validación: El cliente validará en todo momento si la página sigue la guía de marca oportuna.	

Tabla 10. Historia de Usuario (VII)

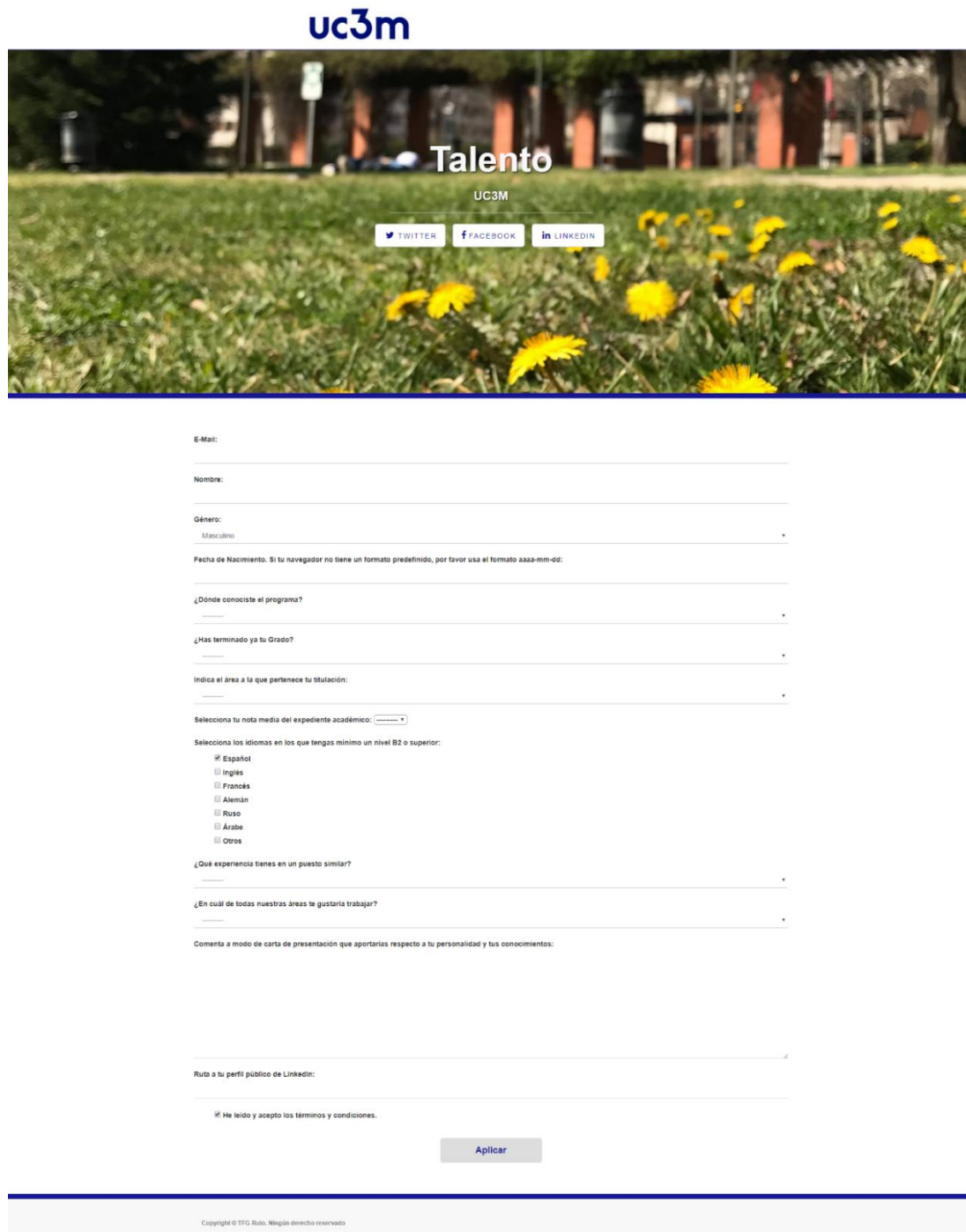
Historia de Usuario	
ID: 8	Usuario: Cliente
Nombre historia: Customización	
Prioridad en Negocio: Media	Riesgo en desarrollo: Bajo
Puntos estimados: 5	Iteración asignada: 8
Programador responsable: Raúl de la Torre Navarro	
Descripción: Al usuario le preocupa la velocidad con la que el formulario envíe la información relevante. Nos pide verificar que la velocidad de guardado de datos, así como el mostrar que los datos han sido guardados satisfactoriamente se guarda con relativa rapidez.	
Validación: El cliente podrá validar la velocidad de nuestra aplicación registrándose en esta.	

Tabla 11. Historia de Usuario (VIII)

4.3.3.1. Incremento

En este *release* la entrega realizada de una parte de nuestro producto o subsistema será la del formulario de nuestra aplicación web, con todas las preguntas definidas con el cliente. Así como la customización oportuna.

Tendríamos así el siguiente aspecto de formulario:



uc3m

Talento

UC3M

[TWITTER](#) [FACEBOOK](#) [LINKEDIN](#)

E-Mail:

Nombre:

Género:

Fecha de Nacimiento: Si tu navegador no tiene un formato predefinido, por favor usa el formato aaaa-mm-dd:

¿Dónde conociste el programa?

¿Has terminado ya tu Grado?

Indica el área a la que pertenece tu titulación:

Selecciona tu nota media del expediente académico:

Selecciona los idiomas en los que tengas mínimo un nivel B2 o superior:

- ☒ Español
- ☐ Inglés
- ☐ Francés
- ☐ Alemán
- ☐ Ruso
- ☐ Árabe
- ☐ Otros

¿Qué experiencia tienes en un puesto similar?

¿En cuál de todas nuestras áreas te gustaría trabajar?

Comenta a modo de carta de presentación que aportarías respecto a tu personalidad y tus conocimientos:

Ruta a tu perfil público de LinkedIn:

☒ He leído y acepto los términos y condiciones.

[Aplicar](#)

Copyright © TFG Ruiz. Ningún derecho reservado

Figura 28. Imagen de formulario completo

Así se verá dicho incremento una vez alojemos la solución en el *hosting* oportuno. Habiendo sido acordadas las preguntas con el cliente y validadas por el equipo de los consultores de RRHH. Este formulario será un formulario tipo, preparado para su despliegue de forma automática, dando el *Software* como un servicio, ahorrando costes para nosotros y bajando el precio para el cliente. Si el cliente quiere otro tipo de formulario en el que se inscriban sus candidatos llevará un coste adicional debido a ser algo que requiere de horas extra de trabajo. Todos estos puntos los detallaremos más adelante.

4.3.4. Tercer Release/Sprint

Enumeramos, a continuación, las historias de usuario del tercer *release* que definirán el comportamiento de nuestra aplicación. Valoraremos el tipo y prioridad de estos.

Podemos leer a continuación una breve leyenda de las tareas importantes a realizar en este tercer *release*:

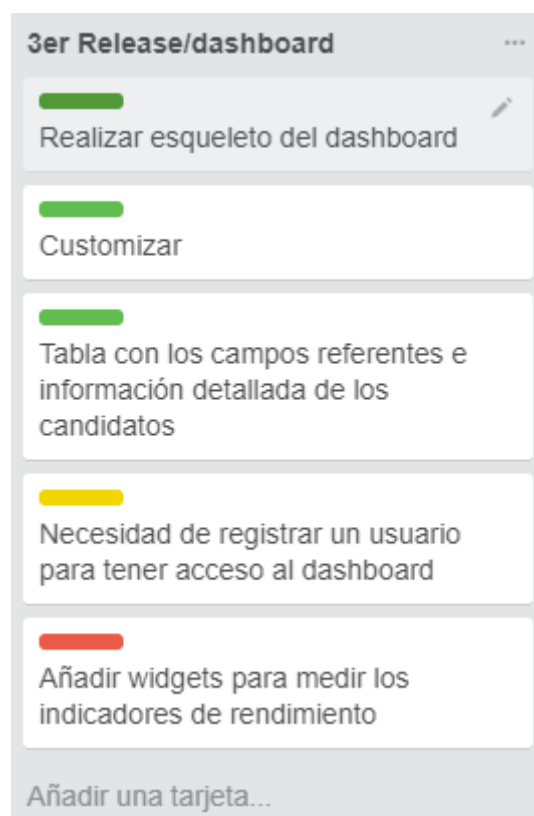


Figura 29. Tareas tercer release

Las tareas están organizadas por orden de prioridad. Como vemos lo más prioritario es realizar el esqueleto del *dashboard* donde irá la información de los candidatos, seguido de las siguientes tareas. Este *release* está prevista su realización en tres semanas de trabajo, 120 horas. Programación, customización siguiendo la guía de marca del cliente, observando posibles mejoras, posibilidad de visualizar los datos en mayor detalle, etc.

1. Realizar el esqueleto del *dashboard* continuando con el esqueleto utilizado en la *Landing Page* como en el formulario.
2. Modificar los estilos del *dashboard* para conseguir una vista de estos moderna y dinámica, así como multidispositivo.
3. Realizar una tabla con los campos y acciones más importantes a realizar sobre un candidato. Como: ver sus campos más relevantes, la posibilidad de ver su perfil público de LinkedIn, etc. Añadiendo en el futuro la posibilidad de obtener un mayor detalle, como por ejemplo resultados de pruebas de evaluación (cuando estas hayan sido implementadas, etc), posibilidad de ver informes customizados de manera online de dicho candidato, etc.
4. En un principio el *dashboard* será alojado en una *url* de caracteres *random* para que solamente los consultores que dispongan de dicha *url* puedan acceder. En un futuro, se implementará un registro de usuarios para que los consultores puedan entrar con un nombre de usuario y una clave.
5. Añadiremos *widgets* para el control de KPI's de nuestra aplicación, deberemos estudiar las posibilidades que tenemos usando la API de GoogleCharts. ("Charts | Google Developers", 2017).

Deberá ser un diseño intuitivo que no conlleve la necesidad de la formación de los consultores de nuestro cliente, para así evitar perder el mayor número de recursos disponibles.

Al terminar el tercer *Sprint* deberemos tener el *dashboard* listo para desplegar y hacerlo visible para los consultores. El registro de los consultores se llevará a cabo por el administrador de sistemas. Teniendo así la tabla con la información de los candidatos lista y los *widgets* para el control de los KPI's, añadiendo en un futuro las posibles mejoras o incrementos que repercutan en esta entrega.

Historia de Usuario	
ID: 9	Usuario: Cliente

Nombre historia: Esqueleto <i>Dashboard</i>	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 25	Iteración asignada: 9
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea un esqueleto con toda la información relevante en el <i>dashboard</i> . Desea que el esqueleto se componga de una cabeza. <i>Widget</i> de las distintas fases del proyecto, <i>widgets</i> para el control de los distintos KPI's y una tabla con la información individual de los candidatos.	
Validación: El cliente nos validará como quiere la distribución del <i>dashboard</i> , en cuanto a organización se refiere.	

Tabla 12. Historia de Usuario (IX)

Historia de Usuario	
ID: 10	Usuario: Cliente
Nombre historia: Customizar <i>Dashboard</i>	
Prioridad en Negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 5	Iteración asignada: 10
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea que la página sea realizada siguiendo la guía de marca de su empresa, utilizando su paleta de colores, su logo, textos específicos para su solución. También desea poder interactuar con las distintas RRSS.	
Validación: El cliente nos informará de la guía de marca que debe seguir el <i>dashboard</i> , no obstante no es una tarea prioritaria.	

Tabla 13. Historia de Usuario (X)

Historia de Usuario	
ID: 11	Usuario: Cliente
Nombre historia: Tabla <i>Dashboard</i>	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 30	Iteración asignada: 11
Programador responsable: Raúl de la Torre Navarro	
Descripción: Se desea una tabla donde poder consultar toda la información sobre dicho candidato. Esta tabla en principio entiende que será básica, sin embargo, irá añadiendo funcionalidades según vayamos completando funcionalidades de las distintas épicas.	

Validación: El cliente verá toda la información relevante a los candidatos en una tabla con información sobre estos

Tabla 14. Historia de usuario (XI)

Historia de Usuario	
ID: 12	Usuario: Cliente
Nombre historia: Registro de usuarios	
Prioridad en Negocio: Baja	Riesgo en desarrollo: Bajo
Puntos estimados: 22	Iteración asignada: 12
Programador responsable: Raúl de la Torre Navarro	
Descripción: El usuario desea tener un control de acceso primario, para que no pueda acceder cualquiera a la información, protegiendo así esta. En el futuro desearía implantar un control de acceso según unos roles asignados a los distintos usuarios con acceso.	
Validación: El cliente podrá registrarse desde una página de registro, solamente podrá acceder a la información si está registrado y validado por los administradores del sistema.	

Tabla 15. Historia de Usuario (XII)

Historia de Usuario	
ID: 13	Usuario: Cliente
Nombre historia: <i>Widgets</i> de KPI's	
Prioridad en Negocio: Media	Riesgo en desarrollo: Alto
Puntos estimados: 38	Iteración asignada: 13
Programador responsable: Raúl de la Torre Navarro	
Descripción: El cliente desea <i>widgets</i> para tener control de los distintos KPI's. Pudiendo obtener así información relevante para su negocio. Mejorando sus procesos.	
Validación: El cliente verá toda la información relevante a los KPI's del proceso en cuestión.	

Tabla 16. Historia de Usuario (XIII)

4.3.4.1. Incremento

Al finalizar este *release*, tendremos una entrega funcional del *dashboard* para el seguimiento por parte de los consultores de RRHH. Muchas de las futuras entregas y posibles mejoras de nuestro producto estarán relacionadas con esta entrega. Tendremos entonces una

interactuación continua con esta parte de nuestro producto para entregar futuras mejoras e incrementos.

4.3.4.2. Gráfica de Avance (Burn Down)

Este apartado mostrará el gráfico del estado de avance del trabajo del *sprint* en curso, en este caso el tercer *release*.

Incluimos este apartado aquí, por considerarlo parte de la planificación de proyecto, ya que se trata del estado en el que nos encontramos actualmente desarrollando.

4.3.4.2.1 Responsabilidades de los implicados

- Responsabilidad del gestor de producto:
 1. Sin responsabilidades específicas, se mantendrá regularmente informado del avance del *sprint* y estará disponible para atender decisiones para la resolución de opciones en *sprints* sobrevalorados o infravalorados.
- Responsabilidades del Scrum Manager:
 1. Supervisión de la actualización diaria por parte del equipo.
- Responsabilidades del equipo técnico:
 1. Actualización diaria del gráfico de avance.
- Responsabilidades del resto de implicados:
 1. Sin responsabilidades.

A continuación, mostraremos el gráfico de *burn down*, la línea azul será la estimación para la terminación del *sprint*, en días (eje x); la línea naranja será el trabajo pendiente en horas (eje y):

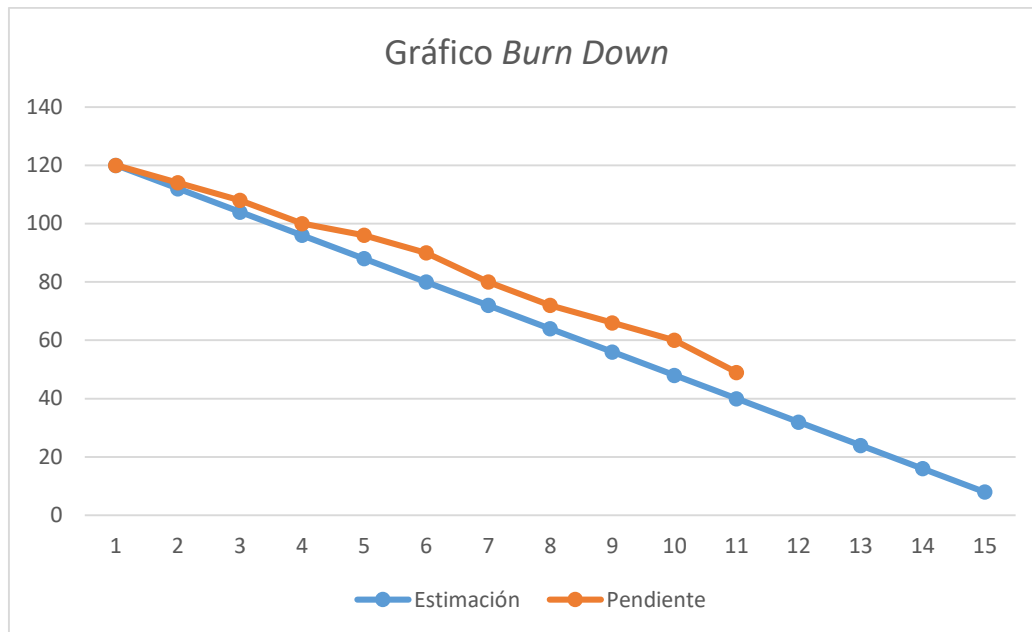


Figura 30. Gráfica de Burn Down

Teniendo así de forma aproximada donde nos encontramos en el tercer *sprint*, así como su estimación prevista para terminación y entrega de incremento.

4.4. Otros puntos de interés sobre la metodología utilizada

Este epígrafe será para explicar los puntos que todavía no han sido explicados sobre la metodología utilizada. Se explica única y exclusivamente por uso didáctico, debido a que en realidad el proyecto está siendo desarrollado por una única persona, sin embargo, se está enfocando para su desarrollo en equipo.

4.4.1. Reunión de inicio de sprint

Esta reunión sirve para determinar las funcionalidades o historias de usuario que se van a incluir en el próximo incremento.

- Responsabilidad del gestor de producto:
 1. Asistencia a la reunión.
 2. Exposición y explicación de las historias que necesita para la próxima iteración y posibles restricciones de fechas que pudiera tener.

- Responsabilidades del Scrum Manager:
 1. Moderación de la reunión.
- Responsabilidades del equipo técnico:
 1. Confección de la pila del *sprint*.
 2. Auto-asignación del trabajo.
- Responsabilidades del resto de implicados:
 1. Sin responsabilidades.

En el proyecto las reuniones de inicio de sprint fueron realizadas con ayuda del tutor de dicho proyecto. No se han cumplido con todas las reuniones, pues serían demasiadas, dichos encuentros plantaron la base para comenzar a desarrollar poco a poco nuestro producto. Antes del comienzo de cada *sprint* al ser una única persona realizando dicho proyecto se paraba para organizar tareas, determinar funcionalidades, preguntar a consultores de RRHH, etc.

4.4.2. Reunión técnica diaria

Puesta en común diaria del equipo con presencia del coordinador del proyector o Scrum Manager de duración máxima de 10 minutos.

- Responsabilidad del gestor de producto:
 1. Sin responsabilidades.
- Responsabilidades del Scrum Manager:
 1. Supervisión de la reunión y anotación de las necesidades o impedimentos que pueda detectar el equipo.
 2. Gestión para la solución de las necesidades i impedimentos detectados por el equipo.
- Responsabilidades del equipo técnico:
 1. Comunicación individual del trabajo realizado el día anterior y el previsto para el día actual.
 2. Actualización individual del trabajo pendiente.
 3. Actualización del gráfico de avance para reflejar el estado de avance.
 4. Notificación de necesidades o impedimentos previsto o ocurridos para realizar las tareas asignadas.
- Responsabilidades del resto de implicados:

1. Sin responsabilidades.

En el proyecto las reuniones técnicas fueron realizadas con ayuda del tutor de dicho proyecto. No se ha cumplido con todas las reuniones, pues serían demasiadas. Todos los días se dedicaba diez minutos para valorar objetiva y críticamente el avance del proyecto en relación a lo desarrollado cada día.

4.4.3. Reunión de cierre de sprint y entrega de incremento

Reunión para probar y entregar el incremento al gestor de producto.

Características de la reunión:

1. Practicas sobre el producto terminado, nunca sobre simulaciones o imágenes.
 2. Tiempo acotado de un máximo de 2 horas.
- Responsabilidad del gestor de producto:
 1. Asistencia a la reunión.
 2. Recepción del producto o presentación de reparos.
 - Responsabilidades del Scrum Manager:
 1. Moderación de la reunión.
 - Responsabilidades del equipo técnico:
 1. Presentación del incremento
 - Responsabilidades del resto de implicados:
 1. Sin responsabilidades.

En el proyecto las reuniones de cierre de *sprint* fueron realizadas con ayuda del tutor de dicho proyecto. No se ha cumplido con todas las reuniones, pues serían demasiadas. Cada vez que terminábamos las tareas de un *sprint* y había que cerrar este y hacer la entrega de este se probaba las funcionalidades indispensables para la entrega satisfactoria de dicho *sprint*.

Capítulo 5. Diseño de la solución técnica

Exponemos en dicho capítulo las funcionalidades de la aplicación analizando de forma detallada la lógica de estas.

Tendríamos así la siguiente arquitectura técnica en nuestra aplicación:

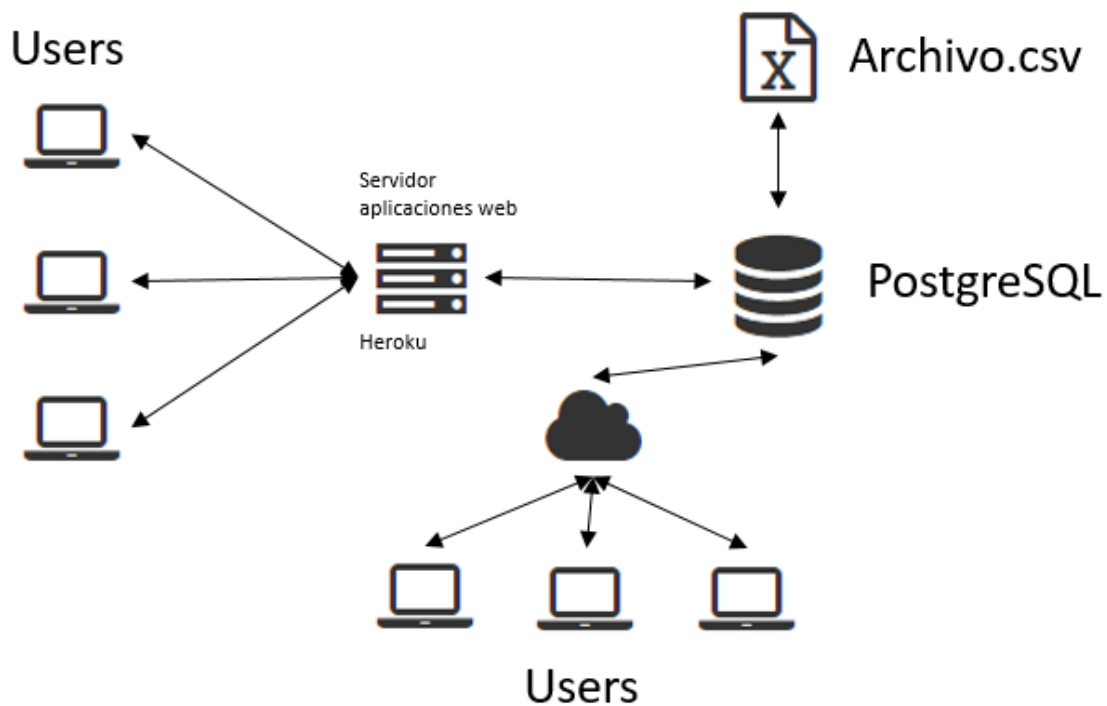


Figura 31. Arquitectura técnica

Tendríamos así por una parte usuarios con rol de consultores que podrían consultar el servidor donde se aloja nuestra solución, consultando la base de datos y pudiendo acceder a toda la información de la página web que hayan rellenado los candidatos. Los candidatos se registrarían a través de una página web siendo sus datos guardados en la base de datos. También los datos obtenidos a través, por ejemplo, de un formulario podrían ser descargados en un fichero.csv, del mismo modo si tomamos la decisión de no tener formulario de inscripción los datos podrían ser cargados desde un fichero.csv. Y estos datos alojados en el servidor al que podrían acceder los usuarios con un rol de consultores.

Tendríamos así el siguiente posible *workflow* de como queremos que funcione nuestra solución técnica:

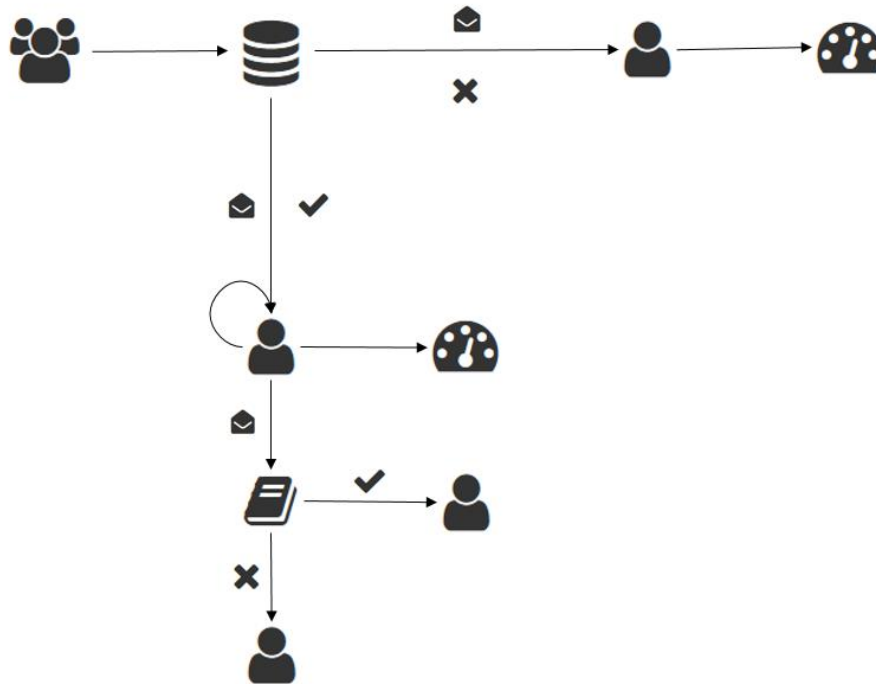


Figura 32. Posible workflow de trabajo

A continuación, explicamos el funcionamiento. Cabe destacar que en el futuro esta lógica será configurable pudiendo alterar, a través de los estados, el flujo de datos.

Los potenciales candidatos, a través del formulario de inscripción entran en nuestra base de datos. Una vez han sido registrados, si el proceso tiene una criba asociada al formulario, entonces tendremos dos posibles casos a estudiar: el primero en el que el candidato no es seleccionado y se le informa a través de un email de *feedback* automático para que conozca su estado, se le incorpora al *dashboard* automáticamente; el segundo caso en el que el candidato si pasa se incorpora automáticamente al *dashboard* y se le envía un email de *feedback* pidiéndole rellenar las pruebas de evaluación consideradas oportunas. Una vez estamos en el estado en el que el usuario ha sido invitado a realizar las pruebas e incorporado al *dashboard*, podrá darse los casos en el que finalice las pruebas, entonces pasará al estado final, en caso contrario permanecerá en el estado anterior hasta cumplir la condición. Una vez el candidato ha realizado las pruebas podemos realizar una selección de forma automática con condiciones o que esta sea realizada por el consultor manualmente. Dado el buen filtrado obtenido de los candidatos una vez finalizado el proceso los consultores preferirán hacer esto manualmente y centrarse más en los candidatos altamente cualificados.

5.1. Bases de datos

Usaremos PostgreSQL para el desarrollo de nuestra solución. Debemos indicar el uso de dicha base de datos en el fichero `setting.py` de nuestra aplicación, de la siguiente manera:

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.postgresql_psycopg2',  
        'NAME': 'ats',  
        'USER': 'postgres',  
        'PASSWORD': 'admin',  
        'HOST': 'localhost',  
        'PORT': 5432,  
    }  
}
```

Figura 33. Base de datos utilizada

Se utilizarán para guardar todos los datos referentes a los candidatos, dichos datos también podrán ser controlados y modificados desde el *BackEnd* de nuestra propia aplicación, así como accediendo a las tablas y modificando desde ahí los datos si fuese necesario realizar algún tipo de cambio importante.

Nuestra Base de Datos procuraremos que siempre la clave primaria sea el uso de un email debido también a que nuestra aplicación tendrá la posibilidad de mandar correos a modo de *feedback* a los candidatos en futuros incrementos para tener la posibilidad de implantar un *feedback* continuado a los candidatos.

La base de datos será definida en el formulario donde se inscriben los candidatos, las claves guardadas en la base de datos vendrán de la fuente de información contenida en diferentes campos. Esta base de datos es definida en el archivo `models.py` de cada una de las partes de nuestra aplicación, así deberemos definir con anterioridad al proyecto nuestro modelo de base de datos, vendrá definido por las preguntas del formulario de inscripción de los candidatos.

A continuación, ponemos un ejemplo del caso en el que obtenemos nuestra base de datos a través del formulario de inscripción de los candidatos.

```

class Candidate(models.Model):
    email = models.EmailField(max_length=50, primary_key=True)
    nombre = models.CharField(max_length=50)
    GENERO_CHOICES = (
        ('Masculino', 'Masculino'),
        ('Femenino', 'Femenino'),
    )
    genero = models.CharField(
        max_length=10,
        default='Masculino',
        choices=GENERO_CHOICES,
    )
    fecha_nac = models.DateField()
    hrconsultant = models.ForeignKey(HrConsultant, null=True, blank=True, on_delete=models.CASCADE)
    jobs = models.ManyToManyField(Jobs, blank=True)
    OTHER_FIND_OUT_PROGRAM_CHOICES = (
        ('LinkedIn', 'LinkedIn'),
        ('Portales de Empleo', 'Portales de Empleo'),
        ('Bolsa de Empleo de la Universidad', 'Bolsa de Empleo de la Universidad'),
        ('Otros', 'Otros'),
    )
    find_out_program = models.CharField(
        max_length=100,
        choices=OTHER_FIND_OUT_PROGRAM_CHOICES,
    )
    BACHELOR_HAVE_CHOICES = (
        ('Si he terminado', 'Si, he terminado'),
        ('No, estudio y >50% créditos', 'No, sigo estudiando y tengo más del 50% de los créditos aprobados'),
        ('No, estudio y <50% créditos', 'No, sigo estudiando y tengo menos del 50% de los créditos aprobados'),
        ('No he estudiado ningún Grado', 'No he estudiado ningún Grado'),
    )
    bachelor_have = models.CharField(
        max_length=100,
        choices=BACHELOR_HAVE_CHOICES,
    )
    BACHELOR_AREA_CHOICES = (
        ('Cs Sociales y afines ADE Derecho Economicas etc Incluyendo dobles titulaciones', 'Cs. Sociales y afines (ADE, Derech'),
        ('Ciencias Exactas o Ingeniería', 'Ciencias Exactas o Ingeniería'),
        ('Ciencias de la Salud', 'Ciencias de la Salud'),
        ('Humanidades', 'Humanidades'),
        ('Otro', 'Otro'),
    )
    bachelor_area = models.CharField(
        max_length=100,
        choices=BACHELOR_AREA_CHOICES,
    )
    BACHELOR_RANGE_ACADEMIC_CHOICES = (
        ('0-4,9', '0-4,9'),
        ('5-5,9', '5-5,9'),
        ('6-6,9', '6-6,9'),
    )

```

Figura 34. Ejemplo de claves en Base de Datos. Models.py

Como podemos ver, cada atributo tiene asignado un *field* el cual después en el formulario será representado mediante distintos *widgets*. También vemos definidas las distintas opciones de respuesta a guardar en cada campo de la base de datos.

En un futuro, para proyectos que no requieran de inscripción a través de un formulario como pudieran ser proyectos con otros fines que no fuesen de selección implementaremos a futuro una funcionalidad que permita subir un fichero.csv. En tal caso la primera fila formará los campos a guardar de nuestra base de datos, siendo la casilla A1 nuestra clave primaria para dicho proyecto.

	A	B	C	D	E
1	email	Competence	Numerical	English	competence_1
2	colymbus@grainsick.net	2	3	9	5
3	submersed@bolection.or	1	1	7	5
4	burlesquely@downshore	8	4	9	5

Figura 35. Ejemplo de claves de Base de Datos. Fichero.csv

Así en este caso tendremos como clave primaria “email” pues es la primera columna de nuestro fichero, para el resto la clave será la primera fila de cada una de las columnas.

5.2. Google Analytics

A continuación, exponemos el porque del uso de Google Analytics, el valor que le aporta el uso de este al consultor de recursos humanos, así como la forma en que tendremos que programarlo para que sea de utilidad para nuestra aplicación.

Usamos así GA porque será de utilidad para el consultor de recursos humanos tener un control de visualizaciones de la *Landing Page* y del formulario, sabiendo siempre que los usuarios están más de un tiempo predeterminado, pues usamos esto para que los datos sean más fidedignos. También nos interesará saber si los usuarios comparten el proceso de selección a través de las distintas redes sociales o si pulsan sobre los iconos de las redes sociales que los redirigen hacia las distintas redes del cliente para el que estamos realizando la aplicación.

El uso de GA nos servirá para después tener unos *widgets* que en el *dashboard* utilizado por el consultor le den toda esta información sin necesidad de tener nosotros que estar pasándola a diario, semanalmente o mensualmente.

Vemos a continuación el código introducido tanto en la *Landing Page* como en la página del formulario.

```

<script>
(function(i, s, o, g, r, a, m) {
  i['GoogleAnalyticsObject'] = r;
  i[r] = i[r] || function() {
    (i[r].q = i[r].q || []).push(arguments)
  }, i[r].l = 1 * new Date();
  a = s.createElement(o),
  m = s.getElementsByTagName(o)[0];
  a.async = 1;
  a.src = g;
  m.parentNode.insertBefore(a, m)
})(window, document, 'script', 'https://www.google-analytics.com/analytics.js', 'ga');

ga('create', 'UA-100383003-1', 'auto');
ga('send', 'pageview');
setTimeout("ga('send','event','Profitable Engagement','time on page more than 30s')", 30000);
</script>

```

Figura 36. Código GA en las cabeceras de Landing Page y formulario

Como vemos deberemos indicar la localización, así como un setTimeout que nos dará un evento si el tiempo que lleva en la página es de más de 30000ms, es decir, de 30 segundos.

Así con este código obtendremos las distintas analíticas que nos puede proporcionar google tanto para nosotros pudiendo mejorar la usabilidad de nuestra aplicación y conseguir cada vez una captación mayor de candidatos, así como para nuestro cliente que podrá mejorar sus procesos en el futuro.

El código que introduciremos en los distintos botones de nuestra página para saber si los candidatos comparten en redes sociales, o si va de la *Landing Page* al formulario será el siguiente:

```

<a href="https://www.facebook.com/uc3m/" onclick="ga('send','
  event','rrss','click','facebook')" class="btn btn-default
btn-lg" target="_blank"><i class="fa fa-facebook"></i> <span
class="network-name">Facebook</span></a>

```

Figura 37. Código Ga en los distintos botones

Así, a través de este código podremos saber los clics que tiene nuestra aplicación en los distintos botones que hemos colocado, sabiendo, así como interactúan los candidatos con nuestra página.

En la vista de nuestro proyecto deberemos crear dos filtros: uno para excluir el tráfico de otros proyectos y que así no se nos ensucien las métricas del presente proyecto y otro para incluir también la dirección del formulario dentro del filtro y así obtener también sus métricas.

5.3. Formulario de inscripción

Como indicamos anteriormente nuestra base de datos dependerá de los campos definidos a guardar en la base de datos. Sin embargo, los *widgets* mostrados en el formulario dependerán del fichero forms.py en el cual estarán definidos.

En el caso de querer que alguna pregunta sea no requerida, así como hacer una pregunta en el que tengamos las opciones predefinidas, lo indicaremos a través de distintos atributos en models.py.

Vemos a continuación un ejemplo de como definimos los *widgets* a utilizar en nuestro formulario de inscripción.

```
fields = [
    'email',
    'nombre',
    'genero',
    'fecha_nac',
    'find_out_program',
    'bachelor_have',
    'bachelor_area',
    'bachelor_range_academic',
    'english_level',
    'other_languages',
    'experience',
    'wish_area',
    'comments',
    'url_linkedin',
    'terms_and_conditions',
]
labels = {
    'email': 'E-Mail',
    'nombre': 'Nombre',
    'genero': 'Género',
    'fecha_nac': 'Fecha de Nacimiento. Si tu navegador no tiene un formato predefinido, por favor usa el formato aaaa-mm',
    'find_out_program': '¿Dónde conociste el programa?',
    'bachelor_have': '¿Has terminado ya tu Grado?',
    'bachelor_area': 'Indica el área a la que pertenece tu titulación',
    'bachelor_range_academic': 'Selecciona tu nota media del expediente académico',
    'english_level': 'Indica tu nivel de inglés',
    'other_languages': 'Coméntanos que otros idiomas conoces, siempre y cuando tu nivel sea un B2 o superior',
    'experience': '¿Qué experiencia tienes en un puesto similar?',
    'wish_area': '¿En cuál de todas nuestras áreas te gustaría trabajar?',
    'comments': 'Comenta a modo de carta de presentación que aportarías respecto a tu personalidad y tus conocimientos',
    'url_linkedin': 'Ruta a tu perfil público de LinkedIn',
    'terms_and_conditions': '¿Has leído y aceptas los términos y condiciones de privacidad? Si no aceptas tu candidatura',
}
widgets = {
    'email': forms.TextInput(attrs={'class': 'form-control'}),
    'nombre': forms.TextInput(attrs={'class': 'form-control'}),
    'genero': forms.Select(attrs={'class': 'form-control'}),
    'fecha_nac': forms.TextInput(attrs={'class': 'form-control'}),
    'find_out_program': forms.Select(attrs={'class': 'form-control'}),
    'bachelor_have': forms.Select(attrs={'class': 'form-control'}),
    'bachelor_area': forms.Select(attrs={'class': 'form-control'}),
    'bachelor_academic': forms.Select(attrs={'class': 'form-control'}),
    'english_level': forms.Select(attrs={'class': 'form-control'}),
    'other_languages': forms.TextInput(attrs={'class': 'form-control'}),
    'experience': forms.Select(attrs={'class': 'form-control'}),
    'wish_area': forms.Select(attrs={'class': 'form-control'}),
    'comments': forms.Textarea(attrs={'class': 'form-control'}),
    'url_linkedin': forms.URLInput(attrs={'class': 'form-control'}),
    'terms_and_conditions': forms.Select(attrs={'class': 'form-control'}),
}
```

Figura 38. Código widgets del formulario

En el futuro podremos utilizar distintos tipos de *widgets* para el formulario, maquetando soluciones que mejoren la experiencia de usuario, como por ejemplo crear un calendario para seleccionar la fecha de nacimiento sin necesidad de escribir a mano.

5.4. Dashboard

A continuación, dividiremos por partes el *dashboard*:

1. *Widget* con las distintas fases del proyecto.
2. *Widgets* con los distintos KPI's definidos para el cliente (en desarrollo), en la sección hablamos un poco sobre lo investigado acerca de la implementación de dichos *widgets*.
3. Tabla con toda la información y enlaces relevantes para obtener toda la información del candidato.

5.4.1. Widget – Fases del proyecto

Introduciremos en el *dashboard* la opción de tener un *widget* de tipo html, este *widget* será un trozo de código estático, el cuál indique a los implicados en el proyecto y con acceso al *dashboard* la información de las distintas fases del proyecto, así como sus duraciones.

Obtendremos así un *timeline* similar a este, con las distintas fases que hayamos definido en el proyecto con anterioridad.

Timeline - Fases del Proyecto

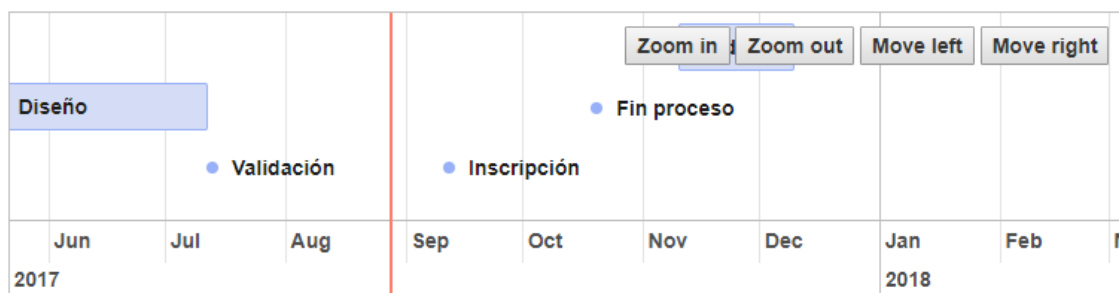


Figura 39. Widget - Fases del proyecto

Las distintas fases del proyecto se definirán como bien dijimos en estático en la página del *dashboard* en formato json de la siguiente manera:


```

<h2>Timeline - Fases del Proyecto</h2>
<div id="visualization">
  <div class="menu">
    <input type="button" id="zoomIn" value="Zoom in" />
    <input type="button" id="zoomOut" value="Zoom out" />
    <input type="button" id="moveLeft" value="Move left" />
    <input type="button" id="moveRight" value="Move right" />
  </div>
</div>
<script type="text/javascript">
  // create a timeline with some data
  var container = document.getElementById('visualization');
  var items = new vis.DataSet([
    {
      id: 0,
      content: 'Análisis',
      start: '2017-05-05',
      end: '2017-05-12'
    }, {
      id: 1,
      content: 'Diseño',
      start: '2017-05-12',
      end: '2017-07-12'
    }, {
      id: 2,
      content: 'Validación',
      start: '2017-07-13',
      type: 'point'
    }, {
      id: 3,
      content: 'Inscripción',
      start: '2017-09-12',
      type: 'point'
    }, {
      id: 4,
      content: 'Fin proceso',
      start: '2017-10-20',
      type: 'point'
    }, {
      id: 5,
      content: 'Feedback',
      start: '2017-11-10',
      end: '2017-12-10'
    }
  ]);
  var options = {};
  var timeline = new vis.Timeline(container, items, options);

```

Figura 40. Definición de fases del proyecto

Desde ahí podremos realizar los cambios necesarios, así como añadir fases, modificar fechas, añadir hitos, etc. Esto será un dato meramente informativo que añadirá valor al consultor para saber en qué fase del proyecto se encuentra en cada momento.

Con el siguiente código obtendremos una barra de tiempo en el iframe, la cual nos proporcionará la hora real y se irá moviendo según la hora en la que nos encontramos. El código que realiza esto es el siguiente:

```
// set a custom range from -2 minute to +3 minutes current time
var start = new Date((new Date()).getTime() - 2 * 24 * 60 * 60 * 1000);
var end = new Date((new Date()).getTime() + 7 * 24 * 60 * 60 * 1000);
timeline.setWindow(start, end, {
  animation: false
});
```

Figura 41. Barra de tiempo

5.4.2. Widgets – KPI's

Después de la definición de fases del proyecto y antes de la tabla donde encontraremos toda la información relevante sobre nuestros candidatos inscritos en nuestro formulario para el proceso en cuestión, introduciremos la sección donde irán incluidos los *widgets* para el control de los distintos KPI's.

Dicha función es una fase del tercer incremento de nuestro proyecto el cual se encuentra en actual desarrollo.

Estudiando las posibilidades de insertar los *widgets* para el control de KPI's, vemos que existen dos métodos posibles para implementar esta solución:

1. Realizar una petición a una dirección con los parámetros en cuestión, al usar una etiqueta "img" en nuestro código mostrará la imagen. Dicho método es muy sencillo de utilizar, sin embargo, sus funcionalidades son limitadas. Prescindiremos de la utilización de dicho método.
2. La segunda solución posible es insertar un código javascript en el lugar de la página donde mostraremos el gráfico. Nos decantaremos por dicho método por ser interactivo, disponiendo de información adicional y teniendo varios tipos de gráficos a implementar.

5.4.3. Tabla con información

En dicha tabla encontraremos todos los campos que rellenaron los candidatos. Accederemos a cada candidato con un bucle "for" creando así una fila para cada candidato.

Tendremos la posibilidad de dirigirnos a su perfil público en LinkedIn en el caso de que tenga y haya rellenado dicho campo en el formulario de inscripción. La posibilidad de acceder al resto de información estará alojada en un *popup* que saltará para cada candidato al hacer *click* en “+ Info” y la posibilidad de eliminar al candidato de nuestra base de datos en el caso de que dicho candidato quisiese que su información fuese eliminada.

Vemos a continuación un ejemplo de la tabla y sus posibles acciones.

Nombre	Email	Género	Fecha de Nacimiento	¿Grado?	¿Términos y condiciones?	Acciones
Raúl	raul.de.la.torre.navarro@gmail.com	Masculino	5 de Mayo de 1992	No, estudio y >50% créditos	Si	<div>LinkedIn</div> <div>+ Info</div> <div>Eliminar</div>

Figura 42. Ejemplo de tabla

Más información relevante ✕

Carta de presentación del candidato:
bdfweofuihweiopvfjioeripvgiopergebdfweofuihwei

¿Dónde conoció el programa?:
Portales de Empleo

Área a la que pertenece tu titulación:
Ciencias Exactas o Ingeniería

Nota media del expediente académico:
6-6,9

Nivel de inglés:
B2 Intermedio Alto

Otros idiomas conocidos donde el candidato tiene un nivel B2 o superior:
dfgsdfgdfgsdfgdfghfhg

¿Qué experiencia tiene en un puesto similar?:
Entre 6 meses y 1 año

¿En qué área le gustaría trabajar?:
Comunicacion

Ruta a su perfil público de LinkedIn:
<https://www.linkedin.com/in/ra%C3%BAI-de-la-torre-navarro-4397b8105/>

Nombre
Raúl

Acciones
LinkedIn
+ Info
Eliminar

Figura 43. PopUp con el resto de información sobre el candidato registrado

**¿Desea eliminar el registro de
raul.de.la.torre.navarro@gmail.com?**

Sí, eliminar

Cancelar

Figura 44. Vista para la acción eliminar a un candidato

5.5. Interfaz de usuario

Punto muy importante y vital de nuestra aplicación. Los programadores comúnmente suelen olvidar el imprescindible requisito de hacer una aplicación sencilla, de diseño visual y comprensión simple. A menudo suelen tener un departamento de diseño gráfico solo para estas tareas.

5.5.1. Estudio de diseño

Antes de empezar, es necesario realizar un estudio de los componentes de la aplicación para comprender el funcionamiento que dará el usuario de ellos.

Interfaz sencilla e intuitiva que no deberá ofrecer dificultad en su uso. Debe ser lo suficientemente simple como para que cualquier usuario pueda utilizar la aplicación sin ningún inconveniente.

Así, la pantalla se diseñará de tal forma que las opciones y botones sean intuitivos a la hora de realizar acciones a través de ellos. Seguiremos así las siguientes reglas:

1. Utilizar una paleta de colores que sea como la del cliente siguiendo su guía de marca.
2. Acompañar a los botones utilizados en la aplicación de iconos que representen de forma clara su función.
3. Tener en cuenta el redimensionado de los elementos para que nuestra aplicación sea multidispositivo.

5.5.2. Landing Page y formulario

El usuario que desee entrar en la aplicación para registrarse y formar parte del proceso de selección primero irá a la *Landing Page* de nuestro proceso, allí se le informará sobre todo lo relacionado al proceso.

Dicha *Landing Page* seguirá con la guía de marca de nuestro cliente, básicamente usaremos su paleta de colores, así como su logo y sus fotografías.

Encontraremos única y exclusivamente los siguientes botones:

1. Botones de redes sociales: los cuales servirán para compartir la información del proceso en diferentes redes, así como poder ver las redes sociales de nuestro cliente.



Figura 45. Botones de RRSS

2. Botón para ir al formulario, el usuario se dirigirá a la página donde podrá registrarse.

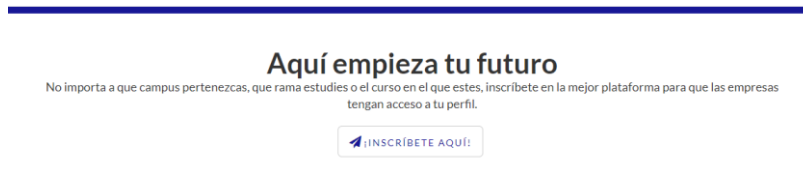


Figura 46. Ir a formulario de inscripción

A continuación, irán al formulario, donde encontraremos los mismos botones de RRSS, así como el formulario de inscripción donde deberán registrar sus datos.

El único botón que tendrán nuevo a diferencia de los distintos *widgets* para responder a las preguntas del formulario, será el botón para enviar la información.



Figura 47. Botón para Aplicar

Una vez hayan aplicado, los candidatos serán redirigidos a una página donde se les informará que su registro ha sido satisfactorio, será posible añadir un mensaje más largo si así lo desea el cliente. Y una vez más, en esta pantalla el candidato también podrá interactuar con los botones de RRSS.



Figura 48. Información sobre registro realizado con éxito

5.5.3. Dashboard

En esta página los únicos usuarios que podrán acceder a ella serán los que tengan o bien su *url* o en futuros incrementos personas que dispongan de un usuario y clave para acceder.

Tendrá exactamente los mismos botones que sus vistas predecesoras, salvando las distancias. Como ya explicamos con anterioridad.

El *dashboard* en un futuro será la parte de nuestra aplicación que tendrá más posibilidades de realizar acciones sobre ella. No obstante, tendrá un diseño que haga que sea intuitivo y fácil de usar por el consultor.

Capítulo 6. Pruebas de los sistemas y manual de usuario

6.1. Pruebas de la aplicación

El estándar seguido para la realización de las pruebas ha sido ISO 9241-11 el cual dice lo siguiente: “la medida en la que un producto se puede usar por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso especificado” (Cancio & Bergues, 2017). Nos hemos centrado así para la realización de las pruebas de usabilidad en el producto. La satisfacción del usuario que usará nuestro producto, el correcto y eficiente desempeño de su trabajo o interacciones con el producto serán los determinantes del grado de aceptación de nuestro producto y así de su usabilidad.

Teniendo así los siguientes procesos que hemos desarrollado para las distintas pruebas en el ciclo de usabilidad:

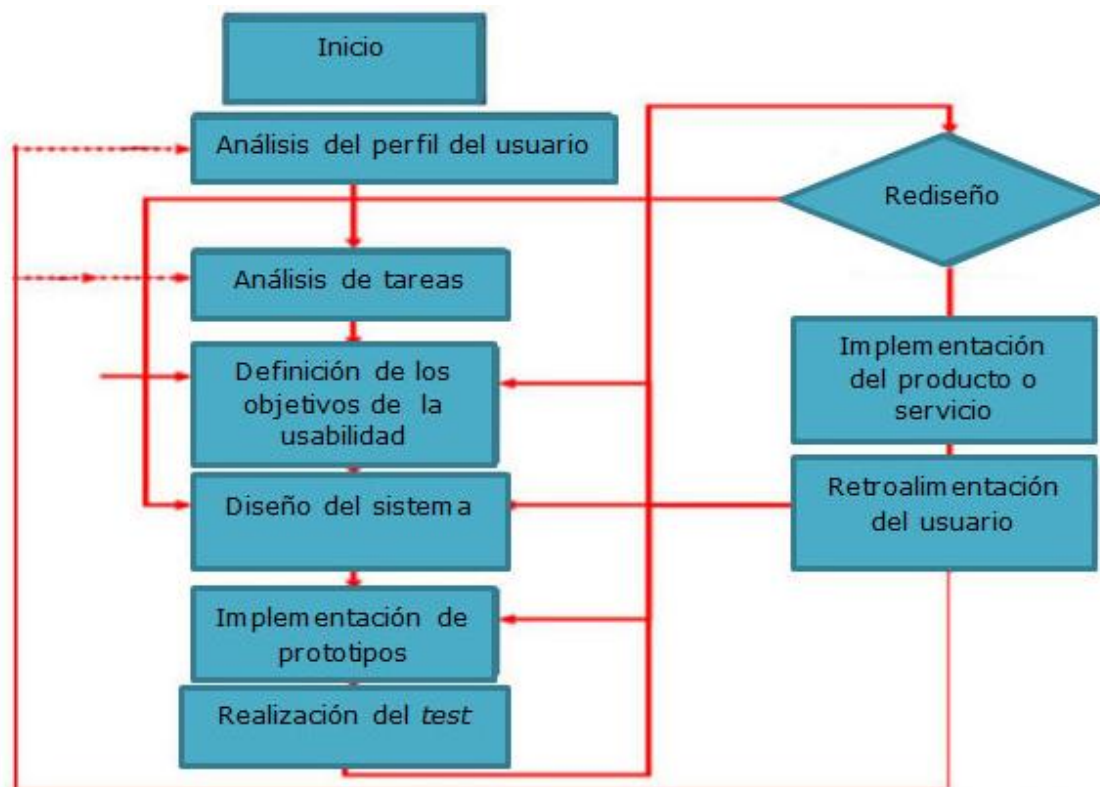


Figura 49. Procesos a desarrollar para la usabilidad

A continuación, expondremos una serie de pruebas realizadas para nuestra aplicación, para cumplir con un mínimo de requisitos. Para una mejor comprensión de las pruebas

realizadas, cada una de ellas estará expuesta en una tabla. Estarán formadas por los siguientes atributos:

- ID: Vinculadas a un identificador único. Usaremos la siguiente nomenclatura: P-XX, donde P significará prueba y XX el número de esta.
- Descripción: Una breve descripción de la prueba realizada.
- Text superado: Marcaremos con una "X" en el caso de que la prueba sea pasada con éxito. Usaremos tres tipos de pantalla, con distintos redimensionamientos, entre las cuales se encontrará una prueba con móviles. Señalaremos con una "X" en el caso en que la prueba haya sido un éxito para esa resolución.

Haremos uso de las siguientes pantallas: pantalla de 27", pantalla de un iPhone 5 y pantalla de un iPad. Observaremos que todas las funcionalidades funcionen de forma adecuada en las diferentes pantallas.

Vemos, a continuación, una tabla que usaremos de plantilla para la realización de las pruebas.

ID	P-XX		
<i>Descripción</i>			
<i>Test superado</i>	[] 27"	[] iPhone 5	[] iPad

Tabla 17. Plantilla de pruebas

Las pruebas realizadas para garantizar un mínimo de calidad son las siguientes:

ID	P-01		
<i>Descripción</i>	<i>Landing Page completamente responsiva.</i>		
<i>Test superado</i>	[X] 27"	[] iPhone 5	[] iPad

Tabla 18. Prueba (I)

ID	P-02		
<i>Descripción</i>	<i>Responsividad del formulario.</i>		
<i>Test superado</i>	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 19. Prueba (II)

ID	P-03		
Descripción	Comprobación de introducción de email válido en el campo email (primary_key)		
Test superado	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 20. Prueba (III)

ID	P-04		
Descripción	Comprobación de estar introducción una <i>url</i> válida en el campo LinkedIn para no introducir cualquier cosa.		
Test superado	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 21. Prueba (IV)

ID	P-05		
Descripción	Comprobación de aviso de campos en blanco en los campos que son obligatorios.		
Test superado	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 22. Prueba (V)

ID	P-06		
Descripción	Velocidad en la que se guarda el formulario y te redirige a la página de registro realizado con éxito es óptima.		
Test superado	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 23. Prueba (VI)

ID	P-07		
Descripción	Responsividad de la página del <i>dashboard</i> .		
Test superado	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 24. Prueba (VII)

ID	P-08		
Descripción	El botón de acción para ir a la <i>url</i> del perfil público de LinkedIn del candidato redirige adecuadamente.		

<i>Test superado</i>	[X] 27"	[X] iPhone 5	[X] iPad
----------------------	---------	--------------	----------

Tabla 25. Prueba (VIII)

ID**P-09**

<i>Descripción</i>	El botón de acción de mostrar más información individual del candidato muestra la información de manera adecuada.		
<i>Test superado</i>	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 26. Prueba (IX)

ID**P-10**

<i>Descripción</i>	El botón de acción de eliminar, elimina de forma adecuada al candidato seleccionado.		
<i>Test superado</i>	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 27. Prueba (X)

ID**P-11**

<i>Descripción</i>	La columna del campo email al pinchar sobre ella hace que salte la aplicación de correo para enviar un email al candidato. Página de <i>dashboard</i>		
<i>Test superado</i>	[X] 27"	[X] iPhone 5	[X] iPad

Tabla 28. Prueba (XI)

6.2. Manual de usuario

Epígrafe destinado a una descripción detallada de la aplicación de ATS, enumerando de forma simple y detallada las acciones a realizar por los usuarios.

La aplicación se presenta para su uso en navegadores web desde cualquier dispositivo con conexión a internet. Sobre todo, para facilitar en la medida de los posible el trabajo a los consultores de RRHH proporcionando un mejor flujo de trabajo y dinamismo en los procesos de selección. En el futuro también influirá en la formación de empleado y así como sesiones de *assessment*.

6.2.1. Landing Page

Será la primera página que el usuario se encuentre al acceder al proceso de selección. Aquí encontrará toda la información relevante al proceso, a la empresa que le seleccionará, etc.

Lo primero que verá el usuario será la cabecera de la página, ahí verá una fotografía de cabecera y, en este caso, tres botones que le redirijan a los perfiles en RRSS de la empresa en cuestión del proceso.



Figura 50. Botones de RRSS

Al pie de página de forma estática hasta aceptar las cookies saldrá el aviso de uso de cookies, una vez aceptado el mensaje desaparecerá. También el usuario podrá ver más información al respecto haciendo clic en dicho mensaje, este le mostrará toda la información relevante sobre el uso de cookies en la página web.



Figura 51. Aceptación y más información de cookies

El usuario verá claramente diferenciadas varias partes de información relevante, acompañadas de una fotografía. Pudiendo tratar esta información, desde lugares de destino hasta centros de formación, pasando por las posibles condiciones laborales.

Campus de Getafe

Situado en la localidad madrileña de Getafe, al sur de Madrid y está dotado de unas modernas instalaciones entre las que destacan sus dos bibliotecas, aulas informáticas en varios edificios, espacios para teledocencia, salas audiovisuales, platós, sala de juicios, zona wifi en todo el campus y un nuevo centro deportivo con piscina cubierta y spa



Figura 52. Información relevante sobre el proceso

Por último y lo más importante, se encontrarán con el botón que le mandará a la página para su registro.

Aquí empieza tu futuro

No importa a que campus pertenezcas, que rama estudies o el curso en el que estes, inscríbete en la mejor plataforma para que las empresas tengan acceso a tu perfil.

[¡INSCRÍBETE AQUÍ!](#)

Figura 53. Ir a formulario de inscripción

6.2.2. Formulario

El usuario que se va a registrar observará la misma cabecera que la *Landing* de la que proviene. Parece un poco redundante, pero podría darse el caso que algún usuario conozca la *url* del formulario y haya accedido sin pasar previamente por la *Landing*.

El usuario deberá seleccionar una opción en las preguntas de tipo “select”, también deberá introducir un email valido, así como *url* de LinkedIn valida en el caso de rellenar dicho campo y rellenar los campos de texto.

Introduzca una dirección de correo electrónico válida.

E-Mail:

sdfgsgfs

Figura 54. Mensaje de introducción de email no valido

Finalmente, una vez el candidato haya rellenado el formulario deberá dar al botón de Aplicar.

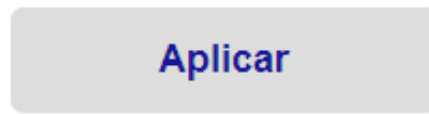


Figura 55. Botón para aplicar al proceso

Entonces el usuario que se registró en la aplicación será redirigido a una página que le indicará que su registro fue realizado con éxito.

Su registro ha sido guardado con éxito

Figura 56. Registro realizado con éxito

6.2.3. Página de dashboard

Esta parte del manual de usuario va dirigido única y exclusivamente para los consultores. Será necesario incluir en el futuro roles de acceso, que puedan variar y depender de cada tipo de proyecto, donde podamos tener distintos tipos de roles. Por ejemplo, un Rol de manager que tenga acceso de toda la información, un rol de departamento que tenga acceso únicamente a los candidatos interesados en su departamento y un rol de línea que tenga acceso a una serie limitada de candidatos, por ejemplo, candidatos cuyos nombres empiecen por la A hasta la F.

Así, habrá que implementar una funcionalidad que nos permita administrar estos roles de forma coherente. A priori, como no es lo más indispensable para dar valor a nuestra aplicación, cualquiera con acceso al *dashboard* podrá ver toda la información relevante.

Una vez que los usuarios se han registrado, sus datos son guardados automáticamente en el *dashboard* para que los consultores puedan realizar acciones sobre estos.

En primer lugar, los consultores se encontrarán con la misma cabecera que el resto de usuarios, por si quisiesen compartir el proceso en las redes sociales y compartir la RRSS de la empresa.

Seguidamente, se encontrarán con un *widget* que marcará las fases del proyecto en un *timeline* y los hitos más importantes de este. Podrán moverse por este *timeline* con el ratón o con los botones para navegar por él.

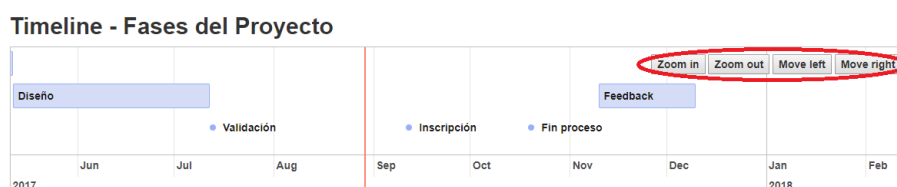


Figura 57. Timeline de proyecto

Después verán la tabla con los candidatos, en la columna del campo email al hacer clic saltará la aplicación de correo de sus ordenadores para enviar un email al candidato.

Tabla - Información individual

Nombre	Email	Género	Fecha de Nacimiento	¿Grado?	¿Términos y condiciones?	Acciones
Raúl	Prueba2@ejemplo.com	Masculino	5 de Mayo de 1992	No, estudio y >50% créditos	Si	LinkedIn + Info Eliminar
Potencia tu mente	prueba3@ejemplo.com	Masculino	5 de Mayo de 2016	Si he terminado	Si	LinkedIn + Info Eliminar
Raúl	prueba@ejemplo.com	Masculino	5 de Mayo de 1992	Si he terminado	Si	LinkedIn + Info Eliminar

Figura 58. Campo email para poder enviar correos a los candidatos

Tendrán los botones para realizar acciones sobre los candidatos. Podrán ir a su perfil público de LinkedIn (si los candidatos rellenaron ese campo), podrán conocer más información sobre el candidato o eliminar a estos.

Tabla - Información individual

Nombre	Email	Género	Fecha de Nacimiento	¿Grado?	¿Términos y condiciones?	Acciones
Raúl	Prueba2@ejemplo.com	Masculino	5 de Mayo de 1992	No, estudio y >50% créditos	Si	LinkedIn + Info Eliminar
Potencia tu mente	prueba3@ejemplo.com	Masculino	5 de Mayo de 2016	Si he terminado	Si	LinkedIn + Info Eliminar
Raúl	prueba@ejemplo.com	Masculino	5 de Mayo de 1992	Si he terminado	Si	LinkedIn + Info Eliminar

Figura 59. Acciones a realizar

En el caso de desear la eliminación de un candidato por completo del proceso, al hacer clic en eliminar el consultor será redirigido a la página de eliminación. Al dar a eliminar sus datos se eliminarán y volveremos al listado, si damos a cancelar sus datos no se borrarán y también volveremos al listado.

¿Desea eliminar el registro de prueba@ejemplo.com?

Sí, eliminar

Cancelar

Figura 60. Eliminar datos de candidato

Capítulo 7. Presupuesto

En este capítulo desglosaremos el presupuesto y hablaremos sobre el coste, beneficios y margen de beneficios de nuestro proyecto, para tener un producto rentable teniendo también la capacidad de ser vendido de forma modular.

Para el cálculo del presupuesto tendremos en cuenta los costes de materiales, de personal y los indirectos. Además, se añadirán los márgenes de beneficio y de riesgo a tener en cuenta. Tendremos un total de tres presupuestos, influirán los costes mencionados antes para cada uno, debido a que cada uno de los presupuestos se relacionará con uno de los incrementos. Esto facilitará saber el coste real de cada funcionalidad, haciendo también posible el contrato de solo una parte del proyecto, si alguna empresa quisiese no disponer de algún modulo.

Dividiremos los gastos por semanas, debido a que los *sprints* y entregas de incrementos se hacen por semanas. Siempre se trabajará por semanas, pudiendo componerse los *sprint* por una única semana o hasta un máximo de cuatro semanas de duración, computando las horas de trabajo dependiendo de las semanas que dure cada *sprint*. A continuación, desglosaremos los gastos comunes derivados de personas, material, etc.

- Lugar de Trabajo: se trabajará desde el lugar de residencia de los desarrolladores, por lo que no tendremos gasto de alquiler, pero si tendremos gasto de: luz, agua, teléfono, conexión a internet y material de oficina. Estimamos en 120 euros por semana de trabajo.
- Ordenador: El coste del equipo de sobremesa utilizado por el único desarrollador en plantilla actualmente es de aproximadamente 2.000 €.
- *Software*: Microsoft Word gratuito, entorno de desarrollo gratuito.

Estimaremos el coste del personal según el número de horas empleadas en el trabajo. Consultando los honorarios profesionales según el COITT, Colegio de Ingenieros Técnicos de Telecomunicación. (2017)

Fijaremos un precio por hora de trabajo de 30 euros. Cada entrega tendrá un coste diferente, lo vemos más adelante desglosado para cada entrega. A este coste habrá que añadir el 21% de IVA, el resto de productos ya llevan contabilizado el IVA en el precio.

1er Release (40 horas)	
Descripción	Coste
Lugar de trabajo	120 €
Ordenador	2.000 €
Raúl de la Torre Navarro	$(30 \text{ €} \times 40 \text{ horas}) + 21\% \text{ IVA} = 1.452 \text{ €}$
Coste total	3.572 €

Tabla 29. Presupuesto primer release

2º Release (80 horas)	
Descripción	Coste
Lugar de trabajo	$120 \text{ €} \times 2 \text{ semanas} = 240 \text{ €}$
Ordenador	No aplica, ya facturado
Raúl de la Torre Navarro	$(30 \text{ €} \times 80 \text{ horas}) + 21\% \text{ IVA} = 2.904 \text{ €}$
Coste total	3.144 €

Tabla 30. Presupuesto segundo release

3er Release (120 horas)	
Descripción	Coste
Lugar de trabajo	$120 \text{ €} \times 3 \text{ semanas} = 360 \text{ €}$
Ordenador	No aplica, ya facturado
Raúl de la Torre Navarro	$(30 \text{ €} \times 120 \text{ horas}) + 21\% \text{ IVA} = 4.356 \text{ €}$
Coste total	4.716 €

Tabla 31. Presupuesto tercer release

A continuación, presentaremos el beneficio a obtener tras la realización de costes. Añadiremos un margen de beneficio del 15% del coste total. El margen de aplicación será calculado para el coste total de cada una de las entregas de producto, este margen es bastante bajo debido a que se irán implementando mejoras de forma gradual según vaya creciendo.

Release	Coste Total	Margen de beneficio	Beneficio total
1º	3.572 €	15%	535.8 €
2º	3.144 €	15%	471.6 €
3º	4.716 €	15%	707.4 €

Tabla 32. Margen de beneficio de los tres release

Además, debemos añadir un margen de riesgo sobre el coste total, sin incluir los beneficios. Fijado en un 5%.

Release	Coste Total	Margen de riesgo	Beneficio total
1º	3.572 €	5%	178.6 €
2º	3.144 €	5%	157.2 €
3º	4.716 €	5%	235.8 €

Tabla 33. Margen de riesgo de los tres release

Ahora presentaremos la cantidad total a cobrar al cliente en cada una de las entregas de producto.

Descripción	Coste
Coste total	3.572 €
Beneficio	535.8 €
Margen	178.6 €
Cantidad Total (IVA incluido)	4.286'4 €

Tabla 34. Cantidad total primer release

La realización de esta entrega de producto tiene un coste de: CUATRO MIL DOSCIENTOS OCHENTAY SEIS CON CUATRO EUROS.



Fdo. Raúl de la Torre Navarro

Graduado en Ingeniería de Sistemas Audiovisuales

Descripción	Coste
Coste total	3.144 €
Beneficio	471.6 €
Margen	157.2 €
Cantidad Total (IVA incluido)	3.772.8 €

Tabla 35. Cantidad total segundo release

La realización de esta entrega de producto tiene un coste de: TRES MIL SETECIENTOS SETENTA Y DOS CON OCHO EUROS.



Fdo. Raúl de la Torre Navarro

Graduado en Ingeniería de Sistemas Audiovisuales

Descripción	Coste
Coste total	4.716 €
Beneficio	707.4 €
Margen	235.8 €
Cantidad Total (IVA incluido)	5659.2 €

Tabla 36. Cantidad total tercer release

La realización de esta entrega de producto tiene un coste de: CINCO MIL SEISCIENTOS CINCUENTA Y NUEVE CON DOS EUROS.



Fdo. Raúl de la Torre Navarro

Graduado en Ingeniería de Sistemas Audiovisuales

Capítulo 8. Conclusiones y futuras entregas

En este último capítulo serán expuestas las conclusiones derivadas de la elaboración de este trabajo, así como las futuras implementaciones que se han detectado que son necesarias por parte de consultores y candidatos.

8.1. Conclusiones

Debido al uso de una metodología ágil hemos podido ir probando la aplicación a medida que íbamos realizando las pruebas pertinentes, así como una retroalimentación con los comentarios recibidos de candidatos y consultores.

Sobre los objetivos, hemos logrado cubrir los objetivos generales del proyecto. Hemos mejorado la eficiencia y accesibilidad de los procesos de selección, realizando una página de uso sencillo y atractiva para el usuario.

Respecto a los objetivos específicos, que a su vez iban ligados a los generales, hemos conseguido realizar el más importante de todos ellos, el que siembra los precedentes de la diferenciación de nuestro producto y que podremos mantener según vayamos implantando nuevas funcionalidades. Estamos hablando de la realización de una web accesible, capaz de ser mostrada en cualquier dispositivo con conexión a internet y usable para los usuarios que interactuarán con ella. El resto de objetivos específicos, solo ha podido valorarse su posible estimación de tiempo que llevará realizarlos.

Durante la realización del proyecto se han afianzado los conocimientos de asignaturas cursadas durante la titulación como Aplicaciones Multimedia y Gestión de Proyectos y Normativa de Telecomunicaciones, a la vez de aprender nuevas metodologías de organización. Evidentemente han sido más asignaturas en las que hemos podido poner a prueba nuestros conocimientos, así como ampliar estos.

Hemos superado retos como aprender una nueva metodología de gestión, partiendo de conocimientos muy básicos sobre esta, así como solucionar problemas relacionados con el desarrollo *software*, a base de investigar y solucionar estos. Muchos de estos problemas, a veces, por su simplicidad resultaban más complicados de resolver que los problemas

complejos, sobre estos últimos su resolución al tenerlos localizados en un punto en concreto es más sencilla.

A destacar, el hecho de haber realizado una aplicación completamente desde cero y funcional para comenzar a funcionar. En el futuro dependiendo del capital que tuviésemos ampliaremos dicha idea y será planteado un futuro emprendimiento empresarial en esta idea, mejorando día a día. Por todo esto, los objetivos del presente proyecto realizado han sido cumplidos en su totalidad, teniendo pretensiones de ampliar en el futuro este proyecto, haciendo especial hincapié en ampliar el equipo de desarrollo para tener una mayor funcionalidad de nuestra aplicación.

8.2. Trabajos Futuros

Son los futuros incrementos que implementaremos debido a haber tenido conversaciones con distintos usuarios de *software* ATS, habiendo interactuado con consultores y candidatos conociendo las necesidades de todos los implicados.

Así, tendríamos las siguientes entregas de incrementos.

8.2.1. Primer trabajo futuro

Necesidad de *feedback* continuado al candidato. Implicará meter una lógica de estado a los candidatos teniendo así un árbol binario de decisión. Aplicaremos, dependiendo de cada proceso distintos tipos de criba que sigan distintos criterios. Sí el candidato no cumple los requisitos mínimos recibirá un email donde se le agradecerá su participación, sino se le enviará a la siguiente fase que dependerá del proceso.

Esto también servirá para otros procesos, como los de evaluación, para mantener informado al candidato si ha realizado las pruebas de evaluación o no, por ejemplo. Dando un valor añadido a nuestra aplicación que hará que los candidatos tengan una mejor experiencia en el proceso.

8.2.2. Segundo trabajo futuro

En esta segunda entrega futura, el consultor deberá poder realizar informes desde la información individual del candidato.

Estos informes deberán de ser multiplataforma, a la vez que si el consultor desea trabajar en formato papel pueda imprimir dicho informe y no tenga inconveniente con estos. En dichos informes aparecerá toda la información pertinente del candidato, así como los resultados de todas las pruebas que el candidato hubiese realizado.

8.2.3. Tercer trabajo futuro

Se implementará meter en el proceso de forma automática la invitación a una realización de pruebas. El objetivo es realizar estas pruebas ya sean de conocimientos o de perfil psicológico, jugando a un juego, mejorando así en todo momento la experiencia de usuario, provocando que dicho candidato desee volver a aplicar a procesos similares, haciendo también que el proceso sea divertido y ameno.

Referencias

Las ventas del software RRHH crecerán entre un 10 y 20% » MuyCanal. (2017). *MuyCanal*. Retrieved 31 August 2017, from <http://www.muycanal.com/2017/01/20/ventas-software-rrhh>

Cómo superar los 'Applicant Tracking Systems' para encontrar empleo. (2017). *Practicopedia*. Retrieved 31 August 2017, from <http://trabajo.practicopedia.lainformacion.com/busqueda-de-trabajo/como-superar-los-applicant-tracking-systems-para-encontrar-empleo-22900>

FileFinder: Executive Search Software | Cloud Executive Recruiting Software | Headhunter Software. (2017). *Dillistone Systems*. Retrieved 31 August 2017, from <https://www.dillistone.com/>

Results: Applicant Tracking Software. (2017). *Softwareadvice.com*. Retrieved 31 August 2017, from http://www.softwareadvice.com/hr/applicant-tracking-software-comparison/?deployment_id=&market_products_sort_order=&market_products_sortby=great_fit&more=true&price_ranges=&stars=&segment_id=434&platforms=&int_site_code=&size_id=%20

Pérez, J. (2017). *Ya está aquí el nuevo Zoho Recruit!!*. *Blog SagitaZ*. Retrieved 31 August 2017, from <http://blog.sagitaz.com/es/nuevo-zohoh-recruit/>

Staffing Software | Applicant Tracking System | Bullhorn. (2017). *Bullhorn*. Retrieved 31 August 2017, from <http://www.bullhorn.com/>

Oracle Taleo Cloud Service | Cloud Talent | Oracle. (2017). *Oracle.com*. Retrieved 31 August 2017, from <https://www.oracle.com/applications/taleo.html>

Recruiting Software Features | Greenhouse. (2017). *Greenhouse*. Retrieved 31 August 2017, from <https://www.greenhouse.io/features>

Online Recruitment Management System Features | Newton Software. (2017). *Newton Software*. Retrieved 31 August 2017, from <https://newtonsoftware.com/features/>

Talent Clue: software de reclutamiento para atraer y fidelizar. (2017). *Talentclue.com*. Retrieved 31 August 2017, from <https://talentclue.com/es>

HTML5. (2017). *Mozilla Developer Network*. Retrieved 5 April 2017, from <https://developer.mozilla.org/es/docs/HTML/HTML5>

CSS3. (2017). *Mozilla Developer Network*. Retrieved 5 April 2017, from <https://developer.mozilla.org/es/docs/Web/CSS/CSS3>

- JavaScript. (2017). *Es.wikipedia.org*. Retrieved 5 April 2017, from <https://es.wikipedia.org/wiki/JavaScript>
- jQuery. (2017). *Es.wikipedia.org*. Retrieved 5 April 2017, from <https://es.wikipedia.org/wiki/JQuery>
- Python. (2017). *Es.wikipedia.org*. Retrieved 17 April 2017, from <https://es.wikipedia.org/wiki/Python>
- SQL. (2017). *Es.wikipedia.org*. Retrieved 17 April 2017, from <https://es.wikipedia.org/wiki/SQL>
- JSON. (2017). *Json.org*. Retrieved 17 April 2017, from <http://www.json.org/json-es.html>
- Sublime Text. (2017). *Es.wikipedia.org*. Retrieved 17 April 2017, from https://es.wikipedia.org/wiki/Sublime_Text
- Trello: el administrador de proyectos definitivo - *Dispersium*. (2017). *Dispersium*. Retrieved 18 April 2017, from <http://dispersium.es/trello-el-administrador-de-proyectos-definitivo/>
- Trello - organización on-line y cooperativa. Tutorial subtulado Español. (2017). YouTube. Retrieved 18 April 2017, from <https://www.youtube.com/watch?v=7XFAAZpQkbM>
- Torres, J., Torres, J., & Torres, J. (2017). *Guía de Trello – Gestor de tareas con Kanban - Xelso Blog | Nuevas tecnologías · Entretenimiento*. Xelso Blog | Nuevas tecnologías · Entretenimiento|. Retrieved 18 April 2017, from <http://www.xelso.com/oficina-en-la-nube/guia-de-trello-gestor-de-tareas-con-kanban/>
- Historia de Scrum. (2017). *Proyectos Ágiles*. Retrieved 14 June 2017, from <https://proyectosagiles.org/historia-de-scrum/>
- Luna, A. (2017). *Primera multa a dos webs españolas por violar las normas sobre cookies. Noticias de Tecnología.El Confidencial*. Retrieved 14 August 2017, from http://www.elconfidencial.com/tecnologia/2014-01-24/primera-multa-a-dos-webs-espanolas-por-violar-las-normas-sobre-cookies_80190/
- Charts | Google Developers. (2017). *Google Developers*. Retrieved 19 August 2017, from <https://developers.google.com/chart/>
- ScrumManager - Inicio. (2017). *Scrummanager.net*. Retrieved 31 August 2017, from <http://www.scrummanager.net/>
- (2017). Retrieved 1 September 2017, from <http://www.coitt.es/res/libredocs/Honorarios.pdf>

Stack Overflow en español. (n.d.). Retrieved September 12, 2017, from <https://es.stackoverflow.com/>

Como mostrar gráficos en las páginas web usando la API de GoogleCharts. Retrieved September 12, 2017, from <https://norfipc.com/web/como-mostrar-graficos-paginas-web-usando-api-google-charts.php>

Cancio, L., & Bergues, M. (2017). *Usabilidad de los sitios Web, los métodos y las técnicas para la evaluación*. *Acimed.sld.cu*. Retrieved 14 September 2017, from <http://www.acimed.sld.cu/index.php/acimed/article/view/405/306>

SUMMARY

We live in a changing world, especially in the world of digital resources with increasingly substantial progress. The third industrial revolution or "digital revolution" is dated from 1950 to 1970 by the introduction of use of computers, digital records and the radical changes what that supposed. Nowadays this change is exponential because the majority of the population can access to these technologies easily.

Therefore we are in time when software is improved as a service for the users, making easier their daily tasks of their professional or personal life. Performing that was previously in-depth learning about the subject and where a task could take hours to get it done in minutes.

We are in a world in continuous development; we are constantly connected with the whole world. Always connected to the Internet with different devices: desktop computers, laptops, mobile devices, etc.

We live in a world that is headed for complete digitization. Everyday we have more and more specific jobs. Jobs that are demanded today and tomorrow won't be. Jobs that do not exist today tomorrow will be the most demanded.

For all that is necessary powerful software for the Human Resource Department. We are in the right direction to get it.

All this causes the following needs and implications:

- To implement the same solution for different processes.
- Differentiate the same solution for specifying it for each selection process.
- Give the feedback to the candidates.
- Need for continuing education.
- Need to a program that serves for selection processes, assessment sessions, development, etc.
- Need to keep up with the labour market and its needs.
- Quick changes.
- Deploy modular functionality.
- Ability for give to customers one functionality or another, depending on the needs of their offer and business.

We will need to provide all these solutions to our customers and users of our tools.

We should be aware of the imminent rise of job offers. We need capture the best talent within a large number of candidates or participants for selection and training processes.

We will be always capturing and forming the best people for reduce the unnecessary expenses of companies by optimizing all these processes.

Next, we quickly summarize the needs of our potential customers, as well as the participants in the different processes, thanks to the continuous intercommunication we have had with different stakeholders.

1. A Landing Page, where you will be told who will do the timely process all over this: functions, tests to be performed, process phases, conditions.
2. The Landing Page should be multi-device, since the accesses will be largely from a mobile device due to its increasing use year after year.
3. Consultants request a form where they can obtain KPI's for their project automatically. This way you will be able to obtain the data of this form, to verify, for example: what level of English have the people who they are applying, their academic level and everything that the consultants need to know.
4. Ability to modify these form questions in a simple way, to be able to sell a greater number of projects and increase our benefits, automating the process of realizing projects and spending less resources in developers, being able to invest in other necessary points.
5. Need to be able to perform an initial screening from this form automatically, so that our client invests as little as possible in the expense of human resources consultants. We will thus be able to emphasize the talent of the process.
6. The consultants require a *dashboard* to work with and to be able to realize candidate rankings in that *dashboard*, selecting the best ones. Be able to see the test results of the candidates and any relevant information.
7. The consultants also demand to be able to make timely reports on the participants of the processes online, being able to print these in paper format.
8. Process participants complain of a lack of feedback on their status in the processes. They need be informed every time about their phase and for this purpose an association of phases will be made to the candidates who depend on certain conditions. The conditions may be modified depending on the process. To be able to make this feedback and make it automatic, the primary key must always be the registered person's email.

9. The candidates require more knowledgeable and psychological tests to perform in addition to the implementation of a game for the evaluation of all this, it will be a very powerful differentiating object.

Also due to the international framework in which we are, we must take into account an extension of our product to other geolocations. Particularly, introducing strongly in countries that are entering step by step and with strength in the digital sphere. So, we have Latin America currently in full technological development with much to do in the field of human resources.

Thanks to the study of the art of Spanish companies that offer ATS solutions, as well as foreign companies. We will be aware that, despite the advances in the world of human resources, there is still room for good differentiation. We will be able to offer a very innovative and powerful solution that differentiates us from the rest, we will make our customers hire the most powerful candidates, and we will also help them to boost the careers of their workers with greater capacity.

This project has been conceived for the university use, job offers for its students. However, with future implementations of new features and continuous improvements, it will serve for staff tracking processes and many other applications. It will not only be applicable for the university, but also to many more companies and administrations.

The application should be automated for its use by any type of client and we can achieve the desired objectives with the minimum number of changes. In this way we can use this solution for any type of client who would like to strengthen their Human Resources Department. This will in turn boost the talent of our clients.

Another very important point to keep in mind will be the communication with a law firm to help us comply with the terms and conditions of service we offer. You need to be able to modify these terms depending on the country and the data requested in each offer, modifying these conditions.

Another very valuable option to consider will be the possibility of receiving a database of our client from another type of source other than a form. A possible alternative is the implementation of uploading through a csv. file. Logically, the Consultant will be able to download the data in the future in case he or she wishes to carry out a more exhaustive study of the data he or she has. In the end, we will be able to receive the data from different types of sources.

Obviously, another differentiable factor is the possibility of selling our product in modules. Just as it has been developed in modules, we will be able to sell it in the same way, lowering the price for our customers and making our product offer much more attractive. We have taken into account the needs of our potential customers who may not always need the complete solution for their processes, thus reducing costs for them.

Here are a few examples of the above:

- In the case of only needing to advertise a process or an evaluation reaching as many people as possible, segmented according to customer criteria it is only necessary hire a Landing Page.
- In the case of a process where only the data of the candidates is needed for a process and to count this process, it is necessary the Landing Page and registration form page. The logic of status and initial selection according to the form will depend on whether our client wishes that in this process the candidates received feedback. Only two of our modules are needed.
- In case you need the whole process, they will contract the use of each and every one of the modules of our product.
- If it is an assessment centre where a certain number of employees will be evaluated, we will need to upload candidate data from an external database, or it would also be valid from a form. But we could ignore that data to reduce costs to the client. In that case, we would only need a *dashboard*, although we could also implement logic states so that the participant knows if the process has finished or not.

The three parts that can be sold separately are listed below:

1. Landing Page shows the information about the process.
2. The form where we can receive data and make an initial selection.
3. Finally, the *dashboard*. Here we can see all the information and perform different actions as well as control different KPI's.

All this will allow us to focus our software as a service, increasing our sales and reducing our expenses, as well as those of our customers. Obviously, in order to offer this service we must have our application in a basic state that we can offer to our clients, modifying only the customization.

On the other hand, once we have automated our entire process of creating applications, we have more time for other tasks. As well as creating an AdHoc application in a much simpler way, we will also be able to continue to offer a special service for customers who want a specific application, with specific questions, specific reports, etc. This way, moreover, obtaining a greater benefit for the realization of this type of projects.

We have seen many of the advantages of using an agile methodology, especially the modularity of our project. A continuous feedback with our clients and with potential candidates. All this makes our product a product with high potential to succeed in the market.

This will allow us to grow as our sales increase, as well as expand our staff, especially developers to work on the implementation of new functionalities and continuous automation of processes. Always growing at the same pace as the labour and professional market and promoting the continued growth of other companies, as well as our own company and our software.

We created a positive first impression, creating an employer brand to introduce the right people; this is the new era of talent acquisition. It is essential to present a well-designed online presence offering a coherent image that reflects our objective. We build the brand that our customers need to attract the best digital talent of these times.

Knowing how to distinguish the impact that social networks have today. They are means of attracting talent. Not only we are based on making life easier for everyone involved in the process, we also position ourselves efficiently in the most important social networks to acquire talent, so we make use of metrics that help us to improve these processes day after day, thus positioning our clients' processes at the top and making interactive technologies, digital and social media strategies, thus connecting with the best digital talent.

To conclude, we generate a digital profile of the candidate, always more attractive than the paper profile, more dynamic and easier to interact with. In this way, we offer infinite possibilities to improve all the processes of our customers' companies interested in our product.

That is why we are faced with the continuous change of the world. In human resources these changes are occurring today. Just a few years ago we were in an "analogical" world, everything was done on paper and by hand. Today we are moving towards the digital, automating processes, optimizing costs, acquiring the best possible talent in the market.

Then we get the expected final result. We provide a dynamic and well-differentiated solution from our competitors. All this will allow us to stand out in the market. Always positioning us from all points of view. We all grow together, reduce costs and get the best talent in the market.

All this added to the continuous improvement of our solution, adding value day after day. A solution that takes care of human resources consultants as well as candidates and participants involved in our processes.

Thanks to the use of agile methodologies due to the need for constant and recurrent changes.

Without further comment, we say goodbye leaving this picture of one part of our solution (head of Landing Page).

uc3m

